50X1-HUM

Page Denied

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

STAT

PRINCIPLES OF DESIGN AND TECHNICAL MEANS FOR THE DEVELOPMENT OF HIGH-SPEED INPUT-OUTPUT DEVICES

By

P. P. SYPCHUK and V. I. DOBROSMYSLOV

(Institute for Scientific Research of Computing Machines)

Moscow, USSR

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

In modern electronic machines the time required for the execution of arithmetic operations, including that for the access to the operator storage, has been considerably reduced to get expressed nowadays in milliseconds.

At the same time the problems of speeding-up the process of input and in particular that of output of information has not been completely solved yet, which deficiency adversely effects the rate of execution of the whole scope of operations.

While elaborating the machine the designers were faced with the problem of making a most appropriate choice of the method and means for input-output of information. The required speed of input of data and output of calculation results may be determined at programming of standard problems to be solved in the machine. In this connection the following can be stated:

If the information input-output does not coincide in time with the execution of calculating operations in the machine, the operational speed of external devices, in view of reducing losses in time, should be close to, or commensurable with, the speed of access to the storage.

Another major demand is the condition that the time required for data input-output be as short as possible and commensurable with the time used up for arithmetic operations. In up-to-date machines the time of access to the operator storage provided with magnetic cores is within the limits of 4 and 20 microseconds.

Consequently, with 6-digit cells the storages may yield as much as 1,000,000 to 2,500,000 characters per second.

The speed of recording and read-out of information from the magnetic tape is in the limits from 10,000 to 90,000 characters per second. The speed of electrical printing devices may attain several thousands of lines per second, or, in case of a multidigit device, up to several scores of thousands of characters per second, Consequently, the speeds for both methods of output are commensurable with the speeds of access to storages. When evaluating external devices from the point of view of the commensurability of the time required for input-output and the arithmetic operations, it is necessary to discern the time needed for solving accounting and statistical problems and that taken by the solution of mathematical problems.

The first case is characterized by a great bulk of input-output information and requires relatively high speeds of external devices.

In the second case, the processing of data introduced into the machine is more time-consuming and the use of input-output devices operating at minor speeds will but slightly increase the total time used up for the solution of the problem.

This is why in the case of high-speed computers used for commercial analysis the provision of electromechanical means at the input and output does not prove reasonable and devices for conversion to magnetic tape are to be applied to.

This statement, however, is not always true for machines used for solving mathematical problems.

In order to reduce the time during which the storage gets switched out of the computing processes for information input-output, it is sometimes feasible to introduce an intermediate storage which will enlarge the scope of use of electromechanical external devices.

Such are the requirements set before the external devices by the computing means and storages. It seems reasonable to evaluate now were it but in a few words the available devices used for input-output of information.

The punched card, as a recording medium has a history covering as much as nearly a hundred of years. The possibility of concentrating on a single punched card all the data relating to one object under account and also the presented hereby convenience in sorting and storing the materials have gained for the punched cards a large scope of use, especially for accounting and statistical purposes.

At the same time, the punched card being a rather massive item, a considerable acceleration of its speed encounters some difficulties. This especially counts for the case where provision has been made for card punches at the machine output, since punching requires periodical stoppages of the card. A technical rate of 150 to 250 cards per minute is likely to be the utmost one in this case. Considerable technical speeds may be obtained in card punches in which a relative rest (repose) of the die-punches system and of the punched card is achieved on account of a swinging die. The obtention of an output of 300 to 350 cards per minute for such punching devices is a matter of the nearest future.

If at the machine input provision is made for punches equipped with photo-electric reading, the limit rates are higher and fall within 1,000 and 2,000 cards per minute.

Though the feed of the punched tape encounters less difficulties than that of punched cards, and the related masses and track lengths of separate members may be adopted rather insignificant ones, the output of information from the machine is usually at a quicker rate than the output in the form of perforations on tapes. At the present time tape punches ensuring an output rate of up to 200 lines (characters) per second are under development.

In case tape punches are used for information input, rates of 800 to 1,000 characters per second may be achieved.

Speeding-up the output of results in printed tabular form, in the event of electromechanical devices used, necessitates, in the majority of cases, the solution of problems bearing upon the paper feed and the reduction of the mass of separate members actuated in the course of printing.

The technical rates of modern electromechanical, alphabetical-numerical printers reach 600 to 1,000 lines per minute.

In case nonmechanical printers are used, the printing speed may be increased up to 2,000-5,000 characters per second.

In addition, it is possible to make use, for input-output of a magnetic taper with preliminary transcription, outside the machine, of information from punched cards and tape to magnetic tape, as well as from magnetic tape to punched cards, punched tape and printed tables. Since at a tape speed of up to 3.7 m/sec. and a density of record up to 25 characters per min the speed of recording and read-out reaches 90,000 characters per second. The above method of output has nowadays found wide application.

It seems worthwhile to consider the applicability, in computing machines, of devices dealt with before (see Table 7).

The technical speed of card punches at the input reaches 900 to 2,000 cards per minute. (In the case of Dematic and Transac models--2,000 cards).

However in the majority of cases computers are encountered yielding an output of 200 to 400 cards per min. The speed of card punches at the output does not exceed 100 to 200 cards per minute. The operational speed of tape punches with photo-electric reading is as much as 850-1000 characters per second. The speed, however, of the same devices at the output in the overwhelming number of machine types falls within 25 and 60 characters per second and only in a few models reaches 200 to 300 characters per second.

The technical speed of printers is as high as 900 lines per minute in the case of the numerical-alphabetical alternative, and 1,200 to 1,500 lines per minute in the numerical version. In most computers there are employed typewriters and teletypes having a technical speed up to 10-20 characters per second.

Having made such a general outline of today's information input-output devices, we may proceed to consider, in a more detailed way, the possibilities of speeding-up the operation in some of these devices.

The electromechanical printers present the following advantages:

a) The characters are being printed on paper not subjected to special pretreatment;

b) The received document requires no further processing;

c) Obtention of clear printing;

d) Simplicity of design.

TABLE I

Item No.	Machine name	Punched Card Devices	Punched Tape Devices	Printers	Teletypes and Type-writers	Magnetic Tape
1.	Big-size Electronic Computer (CM)		Input 1200 Numbers/ Min.	1200 lines/ min.		Storage 4x30,000 numbers.
2.	Electronic computer model "Strela"	Input 1200 numbers/ min. Output 600 numbers/ min.		100 lines/ min.	Keyboard with punch	Storage 200,000 43 binary digits
3.	Electronic computer model "Ural"		Input 4500 numbers/ min.	Rod type printer 19 digits 100 numbers/ min.	Keyboard	Storage having a capacity of 40,000 numbers 36 binary digits
4.	IBM 705	Input: 250 cards/min. Output: 100 cards/min.		150 lines/min. 120 digits	Typewriter	Magnetic tape-to-punched cards and punched cards-to-magnetic tape converters

1	2	3	4	5	6	7
5.	Zebra	Input: 200 cards/min. Output: 100 cards/min.	Input: 200 charact-ers/sec. Output: 50 charact-ers/sec.	300 lines/ min.	Typewriter 10-charact-ers/sec. Teletype-10 characters/sec.	
6.	Transac 2000	Input: 200 cards/min. Output: 100 cards/min.	Input: 1000 charact-ers/sec. Output: 60 charact-ers/sec.	900 lines/min. 120 digits	Typewriter 10-charact-ers/sec.	Punched cards-to-magnetic tape and magne-tic tape-to-punched cards converters
7.	Datamatic	Input: 900 cards/min. Output: 100 cards/min.	Input: 100 charact-ers/sec. Output: 10 charact-ers/sec.	900 lines/min.	Typewriter	Punched tape-to-magnetic tape and magnetic tape-to-punched tape converters.

In electromechanical computers bar and rotation type devices have been made use of. As calculations have shown, in the latter case the great masses of members running at unconstant speeds involve great inertia forces, which limit the speed to within 80 and 150 lines per minute.

At the development of high-speed electromechanical printers, three major kinds of printing were discerned:

a) Matrix type printing;
b) Helix type printing;
c) Printing on the fly, i.e., printing by means of hammers, striking through the paper and tape, against a constantly rotating printing wheel.

The matrix printer, in its simplest form, may be conceived as built-up of bars in a quantity of 7 x 5 = 35 bars per digit.

The inconvenience with such a device lies in its great number of components. Its advantages are the following: the line is being printed with all bars striking simultaneously; the printing speed is independent of the number of characters. The number of components in one digit may be reduced if a linear matrix with five bars is adopted, and the character is printed at each seven strikes.

Matrix printers may attain a printing speed of 900 lines per minute.

In the Helix type printers, the characters in one digit are printed by a rod actuated by some electrical means. The character is shaped of points which are obtained at intersections of the rod with one of the threads of a rotating multithread screw.

In devices operating on the principle "printing on the fly" the hammers strike through the paper end tape, a constantly revolving wheel, bearing engraved therein characters.

The main factor determining the possible technical speed of a printer with a constantly revolving printing wheel is the time during which the hammer remains in contact with the wheel at printing (t contact).

It is worthwhile to determine the relation of $t_{contact}$ to the contact speed ($V_{contact}$) and the weight of the hammer.

At the first approximation, there was determined the contact period of steel balls hitting a plane surface ($T_{max.}$).

The weight of the balls was adopted close to the presumed weight of the hammers.

In Table 2 are given the values of $T_{max.}$ as determined experimentally.

The obtained values of $T_{max.}$ may be checked by calculation, using the following Herz equations, known from the theory of elasticity.

$$T_{max} = 2{,}9432 \frac{a\,max}{V_o} \quad \ldots. \qquad (1)$$

$$a\,max = \left[\frac{5\ V^2}{4k_1k_2}\right]^{\frac{2}{5}} \quad \ldots. \qquad (2)$$

$$Tmax = k\,V_o^{-\frac{1}{5}} \quad \ldots. \qquad (3)$$

At a constant speed of contact and at $m_2 \gg m_i$

$$Tmax \simeq B_2\,m_1^{1/3} \quad \ldots. \qquad (4)$$

$$Tmax \simeq B_3\,R_1 \quad \ldots. \qquad (5)$$

where V_o - is the contact speed

m_1 and R_1 the mass and radius of ball

K_1 and K_2 coefficients depending on the material and the size of the ball

B_2 and B_3 constant coefficients.

In Table 3 are given the results of calculation for T_{max} according to equation (1).

TABLE 2

Duration of impact of steel balls against a steel bar (as measured)

Dia. of ball mm	Weight, gr	Duration of impact, milliseconds						
		485 cm/sec.	443 cm/sec.	396 cm/sec.	243 cm/sec.	284 cm/sec.	198 cm/sec.	140 cm/sec.
13	9.80	37	41	42	42	44	45	-
10.7	5.50	29	32	34	36	38	39	-
7.9	2.25	23	24	25	26	27	28	-
6.8	1.09	16	17	18	19	20	22	-
5.8	0.84	13	13	14	15	16	17	-

TABLE 3

(COMPUTED)

13	9.80	34.9	35.6	37.6	37.4	39.0	41.8	44.8
10.7	5.50	28.8	29.4	30.0	30.9	32.2	34.5	37.0
7.9	2.25	21.4	21.8	22.3	23.0	23.0	25.6	27.5
6.3	1.09	16.8	17.1	17.5	18.0	18.8	20.1	21.5
5.8	0.84	15.4	15.7	16.0	16.5	17.2	18.4	19.7

As may be seen by way of comparison, the results of the experiment come pretty close to the calculations made using the Herz equations, in the case of small steel balls striking a plane surface.

The Herz formulas for the determination of $T_{max.}$ have been obtained for quadratic equations of the unstrained bodies surface and a small region of compression in relation to the total surface of the bodies.

Taking as a basis the research work carried on recently by Staerman I.Y. and N.A. Koltchevsky we may develop the following equation for the determination of $T_{max.}$ in the case of two solids of revolution hitting directly and centrally on condition of $2_n \longrightarrow \infty$; where " n " is the order of equations for unstrained surfaces of colliding bodies.

$$T_{max} \simeq \mathcal{J}\left[\frac{1}{K_1 K_2}\right]^{0{,}5} \quad \ldots . (7)$$

where K_1 and K_2 are constant coefficients.

A higher order of equations for the surfaces of unstrained bodies signifies that a greater tightness of contact has been achieved at the impact.

Consequently, the case of the impact described by the equation (7) comes closer to the impact of the hammer against the type wheel then the event of the ball hitting a plane.

As may be seen from the equation (7) at $n \longrightarrow \infty$ - $T_{max.}$ does not depend from the contact speed V_o.

As another approximation to the determination of the contact time at the impact of the hammer against the type wheel, there was made a calculation according to the formula (7), with the result obtained $T_{max.}$ = 24 microseconds.

The duration of hammer and printing wheel contact ($t_{contact}$) cannot be determined by way of calculation.

Major difficulties arise here for the reason that the field of contact between these two bodies is restricted by the cylindrical surface of the hammer and for such a case the Herz assumption of the small field of contact between colliding bodies is not valid.

Moreover, the calculation is complicated by the fact that the type wheel has not a smooth cylindrical surface, since it bears engraved characters, and also by the hammer striking the type wheel through the paper.

The experimental determination of $t_{contact}$ was made for steel and brass hammers weighing 0.6, 0.8, and 1.2 gr respectively at rates of contact from 485 to 198 cm per sec.

Measurements were made of $t_{contact}$ both for the case of direct impact of hammer against type wheel, and for the case of impact through paper.

In Table 4 are given rounded mean arithmetic values of $t_{contact}$ for a hardened steel hammer weighing 0.84 gr.

TABLE 4.

Contact speed cm/sec.	485	396	343	284	198
Time of contact, milliseconds	36	39	40	39	42

For hammers weighing 0.6 and 132 gr at same changes in the contact speed, $t_{contact}$ correspondingly increases from 26 to 27 microseconds, and from 63 to 66 microseconds.

For cases where a steel ball weighing 0.84 gr hits the type wheel through a sheet of paper and carbon paper, $t_{contact}$ at a decrease of the contact speed from 485 to 343 cm/sec., changes from 55 to 65 microseconds, and in the case of a hammer weighing 1.2 gr--from 75 to 85 microseconds.

The results of measurements of the time during which the hammer remains in contact with the type wheel, have allowed to make the following conclusions:

1) The statement made earlier about the hammer operating at printing in conditions close to those of a resilient impact, has got confirmed.

The time of bodies contact at impact changes in the following way:

	Body weight, gr	Contact speed cm/sec	Time of contact milli-sec.	Note
Collision of steel ball with plane	0.84	485	13	As measured
Collision of two solids of revolution at an initial contact of the order of 2	0.84	485	24	As calculated
Collision of steel hammer with type wheel without paper	0.84	485	37	As measured
Ditto, through a sheet of paper	0.84	485	55	Ditto

2) At the impact of a hammer against the type wheel with no paper medium, there is observed an insignificant, (less than $\sqrt[5]{V_o}$) inverse relation of the contact time, which is in harmony with the equations (3) and (7).

3) A direct relation between the contact time and the hammer weight has been observed.

The energy of the hammer, was determined necessary for obtaining a clear impression of the character on one or several documents.

TABLE 5

Number of impressions	1	2	3	4
Energy in ergs for ensuring clear impressions	25,000	35,000	55,000	70,000

The optimum weight of the hammer with the view of obtaining sufficient energy and lot $t_{contact}$ is equal to 0.7-0.9 gr.

During the time the type wheel turns over the circumference are not bearing any characters (t_{reset}), the following steps should be completed:

a) return of the hammer and armature into the initial positions;

b) paper feed;

c) condenser charging.

The time of return of the hammer into its initial position was experimentally determined as equal to 2.5-5 milliseconds.

The time for charging the condenser may be rather small to last 5-7 milliseconds.

The time necessary for paper feed cannot be less than 10-12 milliseconds.

At investigations and in realized models a rather short operational time of electromagnet has been achieved (0.0-1.2 milliseconds).

Such a short duration of electromagnet action allows for the hammer to develop on the short track (approx. 1.3 mm), a speed attaining as much as 0.4 m/sec. and to obtain an energy of 40,000 to 50,000 ergs which is sufficient for printing several documents.

TABLE 5

Number of impressions	1	2	3	4
Energy in ergs for ensuring clear impressions	25,000	35,000	55,000	70,000

The optimum weight of the hammer with the view of obtaining sufficient energy and lot $t_{contact}$ is equal to 0.7-0.9 gr.

During the time the type wheel turns over the circumference are not bearing any characters (t_{reset}), the following steps should be completed:

a) return of the hammer and armature into the initial positions;

b) paper feed;

c) condenser charging.

The time of return of the hammer into its initial position was experimentally determined as equal to 2.5-5 milliseconds.

The time for charging the condenser may be rather small to last 5-7 milliseconds.

The time necessary for paper feed cannot be less than 10-12 milliseconds.

At investigations and in realized models a rather short operational time of electromagnet has been achieved (0.0-1.2 milliseconds).

Such a short duration of electromagnet action allows for the hammer to develop on the short track (approx. 1.3 mm), a speed attaining as much as 0.4 m/sec. and to obtain an energy of 40,000 to 50,000 ergs which is sufficient for printing several documents.

Moreover, with a short duration of electromagnet action the amplitude of dispersion of the characters in the line is reduced.

When proceeding to a computation of the printer, the following conditions are to be satisfied:

1) Printed characters must not get blurred.

$$V_k \, t \text{ contact} \leqslant 0{,}20 \qquad (8)$$

where V_k – = the circumferential speed of the wheel in mm/sec.

2) A clear print is to be obtained

$$\frac{m \text{ related} \quad v^2 \text{ contact}}{2} > A \qquad (9)$$

where m related – is the related mass of the system connected with the hammer;

v contact – the speed of the hammer at the moment of contact;

A – energy in g/cm. required for obtaining a clear print.

3) Repetitive printing is to be ensured.

The time necessary for the return of the system of mechanical members and electrical components into their initial positions should be less than the t_{reset}, assigned in the print cycle.

From the inequalities indicated before there was developed the equation for the determination of the minimum time required for printing one line:

$$T \text{ lines/min} = t_{contact} \frac{S}{0{,}2} \; Z \text{ characters} + t_{reset} \qquad (10)$$

where S = the space between the characters on the circumference of the type wheel.

$Z_{characters}$ = number of characters.

From the equation (8) there may be calculated the maximum circumferential speed for the type wheel, at which no blurring of characters occurs.

$$V_k = \frac{0.20}{50 \cdot 10^6} = 4000 \quad \text{mm/sec.}$$

From the equation (10) there may be computed the minimum time required for printing one line.

$$T \text{ lines/min} = 50 \cdot 10^{-6} \frac{5}{0,2} 10 + 0,010 = 22,5 \cdot 10^{-3} \text{ sec.}$$

which corresponds to a speed of print equal to 44 lines per sec.

At the computation it was assumed that $t_{contact}$ = 50 microsec. Z characters = 10 t_{reset} = 10 milliseconds.

Obviously, the feasible limit printing speed in a digital printer of the type described earlier is of the order of 30 to 35 lines per sec.

When computing punches have an intermittent card feed, a cyclogram is plotted for the punching of one perforation.

On the cyclogram there is plotted the coincidence in time of three processes: interposer's operation perforation of the card by the punch; and shift of the punched card to the following position.

When a suitable design is adopted, the operational time for the interposer may be reduced to 3-4 millisec. and the punching time to 2-3 millisec.

The motion of the card is impeded not only by inertia forces, but also by friction against brushes. Experiments have shown that in cases where the card was freed of the force of friction against brushes, it could be imparted an acceleration of the order of 280 m/sec^2. This means that under most favorable conditions the card feed requires as much as 8 to 10 milliseconds.

The total duration of the cycle of punching one perforation is 15-18 milliseconds which corresponds to a technical speed of 220 to 250 cards per min.

If instead of the card feed with stoppages a continuous card feed is used and a swinging die is introduced into the reproducer's design, the two first components of the cycle (i.e., the time of interposer's operation and the punching time) will remain unchanged. The time which in the first alternative was spent on the card feed, will be replaced now by the time required for the return of the die and punch into their initial position.

Laboratory researches have shown that the total duration of the cycle in the case of the reproducer having a swinging die may be reduced to 10-12 milliseconds, which corresponds to a speed of 320 to 350 cards per minute.

Nonmechanical Means of Recording

The trend towards an increase of the printer's capacity leads to the utilization of nonmechanical methods of printing. At the present time different nonmechanical physical principles are known, upon which high-speed printing mechanisms are being developed.

1. Electrochemical Method of Recording

According to this method recording is obtained by means of a metal electrode entering into mechanical contact with the surface of paper impregnated with an electrolyte. The other side of the paper contacts with a plate serving as the second electrode.

To the electrodes is applied a voltage pulse. The current passing through the paper provokes electrolysis of the electrolyte with which the paper is impregnated, and also a chemical colour reaction, i.e., a change in the colour of the paper at the point of contact of the electrode with the paper.

An obligatory condition being at the same time a great disadvantage of this method is the necessity of working with humidified paper. With this method of recording there may be used both point electrodes and electrodes having the form of numerals or letters.

There are a goodly number of compositions of electrolytes for paper impregnation, allowing to obtain characters of different colours.

The electrochemical method of printing allows to print at a rate of 100 characters per second and necessitates a 20 to 100 mA. current flowing through the working electrode.

As to the quality of paper used for the electrochemical method of printing the requirements are contradictory: on the one hand, the paper must be sufficiently friable and hydroscopic so as to ensure good humidification necessary for the electrochemical reaction; on the other hand it must be strong enough and have a smooth surface to ensure good contact and ready gliding of electrodes.

2. Spark-Discharge Method of Printing

This method of signal recording or character printing is founded on the phenomenon of electric discharge occurring at high voltage, between two electrodes.

Two main processes are known.

a) Breakdown of a layer of paper placed between two electrodes;

b) Breakdown of a layer of air between two electrodes disposed on one side and close to the paper surface.

In the first case the signal recording is obtained in the form of a through hole with burned edges, and in the second case by darkening of the paper surface in the vicinity of the electrodes.

Electrospark recording, as per method (a), necessitates either the use of very thin and, consequently, rather flimsy paper, or of very high tensions. A rather high power consumption and the occurrence of dispersed breakdown points owing to nonuniform electrical properties of the paper surface make this method unfeasible for matrix type printing.

Electric spark recording according to method (b) involves lower tensions and power consumption as compared with the method (â), but requires periodical regulation and cleaning of electrodes.

The electric-spark method allows to obtain rates of recording of 1000 cycles per sec. and requires the application of tensions of the order of 5 kw.

3. Electrothermal Method of Recording

For this method a special double-layer paper is used. The first layer is made up of highly carburized paper. One side of the paper is covered with a thin light-gray colored film of special composition. Recording is effected on this side of the paper.

The film is current sensitive, as its finest particles burn-out under the thermal action, of the electric current, and reveal the black surface of the carburized paper layer.

The electrothermal paper does not need any treatment after recording, and proves sufficiently stable to the effect of light, heat, and humidity.

At the motion of the electrode over the surface of the electrothermal paper in the intervals between pulses, the electrodes may shift some carbonaceous matter and the surface of the coating between the characters may be polluted.

Thanks to the possibility of obtaining small-sized characters from one electrode (approx. 0.5 mm), matrix type printing of numerals and letters may be adopted.

As the working electrode there may be used tungsten, platinum, or platinum-irridium wire, 0.3 mm in diameter. The electrode should be resilient and provide the required contact pressure.

The main advantages of the electrothermal method are the following:

1. Comparative simplicity of character recording, not requiring further treatment.

2. Great speed of printing (over 1000 c.p.m.).

3. Small size of characters obtained from the electrode, allowing to use matrix type printing of numerals and letters.

4. Comparatively low operation voltages.

The disadvantages of the method are:

1. Comparatively high cost of the paper.

2. Evolving of flame and smoke at printing.

4. Photographic Method of Recording

The essence of this method of recording numerals and letters is the following: the results of the problem solution in the form of electric signals, are directed into a decoder which transforms them from a coded system into a decimal one; subsequently the signals are transferred into a device which forms the light signals in the decimal system.

These signals are photographed on film by means of a special camera, the film is developed, dried, and from the negative a print is made on photo paper with subsequent development, fixation, and drying.

If the film is of high sensitivi y and the formed light signals are well illuminated, enormous speeds of recording on film may be obtained (several thousands of characters and more per second). This allows, in some cases, to achieve synchronous operation of the printer and the computer.

Disadvantages of such method of recording are high costs and long-time procedure for transforming the results of the problem solution into a form convenient for visual inspection. This is why the method has not found wide application.

As examples of utilization of such method of recording may be cited photo printing devices in machines models 59 CM "Strela" as well as some models manufactured by foreign firms.

Further development of the photographic method of recording has become feasible thanks to presently devised methods of dry photo printing. So the introduction of the silk-screen method of printing in combination with a cathode-ray tube of special design, permits to devise printing mechanisms operating at a rate up to 10,000 characters per sec.

Electrostatical Method of Recording

The electrostatic method of recording provides for printing characters on special electrostatic paper, this being common paper covered on one side with a dielectric layer.

The process of printing by this method is in three stages.

1. Deposition of an electrostatic charge onto the paper surface by means of electrodes to which a certain potential is applied. The electrodes may be disposed so as to make up a matrix, or a row, but may also be made in the form of numeral or letter types, etc. The potential applied to the electrodes generates electric charges on the paper surface, the arrangement of the charges corresponding to the latent image of a character.

2. Development of the latent image of the character by means of colored resin powder having an opposite charging current and sticking to the paper surface at the points of charges.

3. Fixation of the character image by heat treatment, at which the particles of the colored resin powder are melted, forming a mechanically strong and light proof image.

Since in the course of printing there is no contact between the paper and the electrodes, the electrostatic method of recording may be characterized as a non-contact method of printing, to its great advantage in comparison with the electrochemical or the electrothermal method. Moreover, this method allows to obtain rather high. speeds of recording and has a great power of resolution.

We have worked on the development of printers designed on the latter principle.

The electrostatic printer comprises the following main units:

1. Electronic circuit converting the binary-decimal code into a certain sequence of pulses reaching the printing unit;

2. Printing unit, made up of cylindrical electrodes arranged in a row;

3. Developing unit;

4. Fixation unit;

5. Tape feeding mechanism

The electronic circuit is made of a decoder at the output of which a certain combination of pulses is obtained, ensuring the printing of characters by way of matrix type printing.

The design of such a circuit, essentially, does not meet any difficulties and may be realized by different well known methods.

Pulses obtained at the decoder output are amplified and directed into the printing unit.

The number of electrodes in the printing unit depends on the adopted form of character scanning and of the quantity and arrangement of the characters. The most expedient form of character scanning ensuring the obtention of a clear print and a minimum number of scanned elements is 5 x 7.

In the development unit the fields of the tape having electrostatic charges are being covered with a developing powder.

Researches have shown the expediency of using a developer consisting of two components:

a) coloured resin;
b) quartz sand.

This method of development is based on the tribo effect which consists in the formation of electric charges at mutual friction of small particles of different matter. The tested tribe-electric couples have allowed to obtain electrostatic charges with a sign ensuring negative energizing of quartz sand and positive energizing of resin powder. At the passage of the tape bearing electrostatic charges on its surface through such a composition, particles of coloured resin having a positive charge, under the action of the field of negative charges on the tape, come to stick at the points of these charges.

Tests conducted with this method of development have shown that the quality of the numeral image obtained is quite satisfactory. Moreover, the possibility of using a loose mixture not liable to turn into powder, allows to considerably simplify the design of the developed unit.

The character image obtained by this method has proved unstable and could be easily rubbed off the paper. Therefore it needs fixation. A heating element serves for the fixation of the image.

The 12-digit electrostatic printing device developed at the Institute for Scientific Research of Computing Machines ensures a printing rate of 1000 numerals per second.

With these few considerations on the electrostatic printing device let me conclude the brief outline of some of the design principles and technical means to be used for the development of high-speed input-output devices.

METHODS OF LOGICAL, RECURSIVE AND OPERATOR ANALYSIS AND SYNTHESIS OF AUTOMATA

By

Yu.Y. BASILEVSKY, Yu.A. SHREIDER, I.Y. AKUSHSKY,

Institute for Scientific Research of Electronic Mathematical Machines, Moscow, USSR.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

An automat may be regarded as an operating simulator of a system interacting with the environment and, consequently, as a simulator of a discrete control process. The study and solution of the problem of simulating a real system is a complex process, which involves: the selection of essential characteristics of the system and its interaction with the environment; the approximate description of the system by means of the totality of characteristic features, either algorithm, or structure; the analysis of the obtained information and its transformation into an optimum system of logical functions; the synthesis of a real automat using a certain system of physical components, executing logical functions.

Another alternate approach of the problem is the simulation of a real system (or control process) in the form of a process in an automat with a preset structure, endowed with many degrees of freedom.

In this case, it is desirable that the analysis of the system data be reduced to a certain optimum algorithm and dismember it into independent blocks executed in the automat by means of logical circuits or a program.

The development of digital computer logics and the employment of the machine for handling the information belong to problems of this kind.

The solution of the problem of analysis and synthesis of automata may be performed on the basis of different formal descriptions.

The present paper deals with three alternate approaches to this problem: the calculation of logical functions of time, the analytical medium for the presentation of recursive functions and the operator representation of the process in the automat.

Para.1. AUTOMATA AND LOGICAL FUNCTIONS OF TIME

As an analytical medium for the presentation and analysis of logical features we use the calculus of logical time functions (1), pressupposing that the processes of data conversion occur discretly in time while their description is related to the current moment, the origin.

From here takes its source the method of the process prehistory description, expressed in terms of time delays. To the zero operations on two-valued variables is added the operation of time shift ($D^{\kappa}x$) and relation of logical inequality ($x \leq y$, $x < y$).

In relation to any formula, built-up of two-valued variables by means of introduced operations, the time shift responds to the distribution law.

Over words of equal length, composed of two-valued variables are introduced interdigit operations of negation ($\bar{X}$) crossover ($X \cap Y$) conjunction ($X \cup Y$) and time shift ($D^{\kappa}X$). Besides, the operation is used for cycle shift of the word over a preset number of digits to the left, or to the right ($P_o^{\kappa}X$) and of the relation of logical inequality ($X \leq Y$, $X < Y$). In relation to any formula, built up of words of equal length by means of introduced operations, the time shift and cycle shift respond to the law of distribution.

For characterizing the common features of the word components, are used the operations of conjunction and disconjunction convolution of the word ($L^{\wedge}X$, $L^{\vee}X$).

The time function f (X) from n two-valued arguments $x_1, \ldots, x_n$ composing the word X , may be generated into the normal disconjunction form and expressed in operator form by means of the generating word B , with a length of 2^n , according to the definition

$$f(X) \sim R^{\vee}(B, X) \sim \bigvee_{j=0}^{2^n-1} [L_j B \wedge L^{\wedge}(P_j \cap X)]$$

Each formula ("the original") F ($x, y, \cdots$) on two-valued variables may be compared to the formula ("image") F^* ($X, Y, \ldots$) on words of equal length; The structure of this formula is the same as that of the original, but instead of the time shift operations it contains word cycle shift operations. It may be demonstrated that for any function F over time polynomials from X , not containing time shifts, the distribution law is in force.

$$F[R^{\vee}(B_1, X), \ldots, R^{\vee}(B_{\kappa}, X)] \sim R^{\vee}[F^*(B_1, \ldots, B_{\kappa}), X]$$

Thus, to a great extent the transformation of functions from two-valued variables, may be reduced to the transformations of the generating words of their normal polynomials.

At the description and analysis of the features of discrete processes, the logical functions are often formed in an implicit form of logical time equations, of the kind

$$F(Z) \sim R^{v}(B, Z) \sim c_0$$

where the components of the word Z are two-valued variables $\alpha_1, \dots, \alpha_\kappa$ and their time shifts $D^{\circ}\alpha_\pi, \dots, D^{m-1}\alpha_\pi$, a c_0 is the logical constant (zero).

Such an equation in implicit form defines in the general case the family of logical functions in explicit form

$$\alpha_\pi \sim f_\pi(Z_\pi) \sim R^{v}(B_\pi, Z) \qquad \pi = 1, \dots, \kappa$$

where Z_π does not contain $D^{\circ}\alpha_\pi$. The general method of solution /2/ of time logical equations (the reduction method) leads to a set of inequalities for the generating words of the B_π polynomials of the functions to be determined

$$K_\pi \leq B_\pi \leq M_\pi, \qquad \pi = 1, \dots, \kappa$$

where K_π and M_π are functions of B and of the adopted order of variable reduction. Depending upon the degree of redundancy of the initial equation and of the order of variable reduction, the limits of admissible values for each B_π may be more or less narrow. Thus, for each particular solution is defined the character of the variables and the degree of their dependence from the other variables. In all cases, when solving the equation, we obtain a set of functions corresponding to correctly organized logical nets.

Let us consider the methods of possible mappings between sets of time logical functions and sets of recursive functions.

Take, for instance, a set of time logical functions $z_1, \dots, z_n$ from two-valued variables $x_1, \dots, x_m$. Let, for each two-valued variable, the " u " numerical value be $N(u) = 0$ or 1; besides, we determine for elementary operations

$$N(\bar{u}) = 1 - N(u), N(u \cap v) = N(u) \cdot N(v), N(o^{\kappa} u) = 0^{\kappa} N(u)$$

Attributing to the words

$$X = \Gamma_{i=1}^{m} x_i \quad \text{and} \quad Z = \Gamma_{i=1}^{n} z_i$$

concrete weighting position functions

$$N(X)=\sum_{i=1}^{m} f_x(i)\,N(x_i),\quad N(Z)=\sum_{i=1}^{n} f_z(i)\,N(z_i)$$

we obtain the mapping of the set of time logical functions in the form of a special set of recursive functions defining the number $N(Z)$ as the function of the number $N(X)$ and some precedent values of the numbers $N(X)$ and $N(Z)$.

In this manner may be formed many different numerical interpretations of the given set of logical functions.

The mapping of a given set of recursive functions on a certain special set of logical functions also allows to obtain many solutions depending on the adopted arithmetic system, the limits of modification of numerical parameters and the auxiliary time conditions set for the formation of new parameter and function values.

No general method of transfer from the numerical recursive function to logical functions, has apparently yet been developed.

However, in many cases it is possible to specialize the record of a preset algorithm in recursive form so that only a standard set of recursive functions-components is used, to which a set of standard time logical functions corresponds. In these cases, the logical function and the circuit for the execution of the given algorithm may be easily formed.

As known, this method of approach to the problem is used in designing of machines for information processing.

Now let us consider the conception of the automat and describe it by means of logical functions of time.

Take a square q-dimensioned matrix $\|x_{ij}\|$ with components which are the time polynomials of a predetermined set of external arguments $U_1,\dots,U_K$. The matrix is called true, when each filled column possesses the feature that any column element is a complement to the sum of the rest of the elements with respect to 1.

We employ the true matrix $\|x_{ij}\|$ for defining the finite automat with a "q" states $y_1,\dots,y_q$, assuming that the current value of the matrix element "x_{ij}", determines the transition from the state y_j to the state y_i

with a time delay equal to one unit, according to the equation:

$$y_i \sim \bigvee_{j=1}^{q} [D' y_j \wedge x_{ij}], \quad i=1,\dots,q$$

If at a certain moment only one of the variables $y_1, \dots y_k$ was equal to 1, this feature remains unchanged at any subsequent moment. This means that in the mentioned conditions the set of states is full, and the states themselves are incompatible in pairs; such an automat possesses the properties of single-value and continuity of transfers.

The output functions of the automat $z_1, \dots, z_p$ should be, naturally, defined as polynomials of the current values of the state; in view of the incompatibility of these states in pairs, these functions are reduced to some disjunctions of the variables $y_1, \dots, y_q$.

The aim of the described conception of the automat is to isolate from the automat structure, the primitive (containing no feed backs) functions of external arguments.

If the automat is determined by a transition matrix, its logical structure is directly determined by the afore-indicated set of time logical equations and an adequate correctly organized logical net. The solution of the reverse problem of designing a matrix of automat transitions for a predetermined set of time logical functions, involves the substitution of variables, the solution in relation to new variables of generalized equation in implicit form and the subsequent determination of the matrix elements. Various alternative solutions of the generalized equation lead to different automata which are equivalent in that the same set of output functions corresponds to them.

From each given automat may be formed many equivalent automata either by splitting some states, or by duplicating the group of coupled states (cells). The inverse process consists in the compression of the automat structure by bringing together separate states or cells, on condition that the output function is conserved. This results in a certain minimum structure of the automat / 3 /.

It may be anticipated that the further development of the methods of equivalent transformation of automata structures will be one of the efficient methods of approach to the analysis and synthesis of classes of automata endowed with certain particular features.

Para.2. AUTOMATA AND RECURSIVE FUNCTIONS

Let us consider the description of the automat as a device for the realization of a set of recursive functions. Such a description is especially convenient for automata realizing computing algorithms. We assume that the automat comprises a set of digit registers (cells): $\alpha^1, \alpha^2, \ldots, \alpha^m$ each of which, at a given step of operation, contains a certain number designated further as $[\alpha^i]$ or $[\alpha^i]_n$ and where n - is the number of the step. The state of the automat is determined by the set of numbers $[\alpha^1], [\alpha^2], \ldots, [\alpha^n]$.

We shall assume that part of the registers $\alpha^{m+1}, \alpha^{m+2}, \ldots \alpha^N$ are input registers, i.e., are such the content of which is determined by the information coming from outside. The state of all other registers at the $n+1$- step is determined by the contents of all the registers at the preceding n -step,

$$[\alpha^i]_{n+1} = \mathcal{F}_i([\alpha^1]_n, [\alpha^2]_n, \ldots, [\alpha^m]_n, [\alpha^{m+1}]_n, \ldots, [\alpha^N]_n) \quad (1)$$

$$i = 1, 2, \ldots, m$$

The set of functions $\mathcal{F}_i$ characterizes in full the automat structure. The task of the synthesis is to design, taking as a basis a certain class of algorithms - an automat with such a set of determining functions $\mathcal{F}_i$ which should permit to realize these algorithms / 4 /.

Let us analyze what does represent the set of determining functions $\mathcal{F}_i$ for a three-address computer operating on the position code principle.

Let α^1 - be the register storing the command to be executed,

α^2 - the register storing the address of the command which will be executed in the next step, and $\alpha^3, \alpha^4, \ldots, \alpha^m$ - the operational memory cells.

We do not deal with the input registers considering that at the initial moment all the input information enters into the operational memory.

During each step, there is to be modified only the contents of the cell corresponding to the third address of the command to be executed. For the case when the contents of a certain cell α, is a command, we shall adopt the following designations.

The part of the α cell contents, which is the code (symbol) of the command, we shall designate by $[\alpha]^{ck}$, and the parts $[\alpha]$ which are the first, second and third addresses we shall designate by $[\alpha]^{I}$, $[\alpha]^{II}$ and $[\alpha]^{III}$ respectively. Then we introduce the function $S(a,b)$ which is equal to zero at a $a \neq b$, and equal to 1 at a $=a \cdot b$, and the function $\bar{S}(a,b) = 1 - S(a,b)$. If the executed command is not a command of control transfer, the set of relations (I) may be written as follows

$$[\alpha_1]_{n+1} = [[\alpha_2]_n]_n ,$$

$$[\alpha_2]_{n+1} = [\alpha_2]_n + 1, \qquad (2б)$$

$$[\alpha]_{n+1} = [\alpha]_n \bar{S}(\alpha, [\alpha_1]_n^{III}) + S(\alpha, [\alpha_1]_n^{III}) \sum_{\gamma,\beta,\varphi} \Phi([\gamma]_n, [\beta]_n) S(\gamma, [\alpha_1]_n^{I}) S(\beta, [\alpha_2]_n^{II}) \cdot S(N\Phi, [\alpha_1]_n^{ck}) \qquad (2c).$$

We assume that the three-address command is executed in the following manner: $\Phi([\alpha]_n, [\beta]_n) \longrightarrow \gamma$ where $\varphi(x,y)$ designates an arbitrary operation executed with the numbers x and y during one step by the computer arithmetic device, and $N\varphi$ - the code (symbol) of the corresponding command. In the sum (2B) only one addend differs from zero, for which $\alpha = [\alpha_1]_n^{I}$ and $\beta = [\alpha_1]_n^{II}$, i.e., the operation $\varphi(x,y)$ is executed with data stored at the addresses determined by the command in the register α_0. The presence of summation in all the pairs of cells, (α,β) according to the formula (2B), involves the necessity of full scanning in the memory for selecting the required cells. Let us consider the case when an execution of the control transfer command is possible.

Let us designate by $\Phi_0 \alpha\beta\gamma$ the command of conditional transfer, which has the following meaning: if $[\alpha] > [\beta]$ then the control is transferred to the following by order command, while if $[\alpha] \leq [\beta]$, the control is transferred to the command in the cell γ. Similarly let us designate the command of unconditional transfer by $\Phi_1 \alpha\beta\gamma$ which in all cases transfers the control to the command $[\gamma]$.

At the presence of these commands, the equality (2a) is kept, while the equalities (2б) and (2c) are complicated in the following way. (2б) is replaced by the equality

$$[\alpha_2]_{n+1} = \{[\alpha_2]_n + 1\} S([\alpha_1]_n^{ck}, \Phi_0)\{1 - \mathrm{sign}([\alpha_1]_n^{II} - [\alpha_1]_n^{I})\} + \{[\alpha_2]_n + 1\} \bar{S}([\alpha_1]_n^{ck}, \Phi_0) S([\alpha_1]_n^{ck}, \Phi_1) + \{S([\alpha_1]_n^{ck}, \Phi_0)\,\mathrm{sign}([\alpha_1]_n^{II} - [\alpha_1]_n^{I} + S([\alpha_1]_n^{ck}, \Phi_1)\}[\alpha_1]_n^{III} \qquad (2б')$$

In order to suitably modify the formula (2c) we designate, for abbreviation, the right-hand part of the equation (2c) as G_α.

Then in the considered case we shall have

$$[\alpha]_{n+1} = G_\alpha \{1 - S([\alpha_1]_n^{ck}, \Phi_0) - S([\alpha_1]_n^{ck}, \Phi_1)\} + \qquad (2c)$$

$$+ [\alpha]_n \{S([\alpha_1]_n^{ck}, \Phi_0) + S([\alpha_1]_n^{ck}, \Phi_1)\}$$

This means that if the command $[\alpha_1]$ is a transfer command, then the contents of all memory cells remain as before, i.e.,

$$[\alpha]_{n+1} = [\alpha]_n$$

If the machine contains group operations and other control operations, the set of equalities (2) becomes even more complicated. For the computing algorithm preset in the B operator form [5] it is possible to form a set of functions (I) which realizes this algorithm. This representation is convenient in that the set (I) is directly connected with the structure of the computer (automat) and at the same time is B of "big-block structure". This means that the recursive recording of the automat dynamics does not take into consideration, for instance, the peculiarities of the structure of single digit adders or the methods of multiplication acceleration, while the description of the automat in terms of time logical functions would essentially include these features.

In terms of functions $\mathcal{F}_i$ it is possible to characterize the complexity of the algorithm as well as of the automat realizing this algorithm. (Recently A.P. Ershov (6) has indicated the relation of computing algorithms with the recursive functions).

The complexity of the automat is characterized by the complexity of the set type (2), i.e., by the number of registers with simultaneously changing contents, the number of registers determining the modification of the contents in each register etc. The complexity of the algorithm is characterized by the complexity of the automat realizing the given algorithm during a certain number of steps.

If the system set of recursive relations (1) is expressed in the form of equalities of the type (2), it is possible to build up a set of time logical functions realizing this set of relations. It is only to be remembered that at the transfer to time logical functions it is sometimes expedient to reduce the time scale.

For computing machines a very important point is the choice of an efficient structure of the control device, ensuring a maximum capacity at the given high speed operation of the arithmetic circuits. With this aim in view it is necessary to design such circuits that may conveniently realise complicated combinations of operations for the conversion resetting and formation of addresses. Let us consider the class of recursive functions (I) in which the functions $\mathfrak{F}_i$ are combinations of the following basic functions:

I. Function $S(x,y)$ determined by the equality

$$S(x,y) = \begin{cases} 1 \text{ at } x = y \\ 0 \text{ at } x \neq y \end{cases}$$

2. The function $\mathrm{sign}_1 x$ and $\mathrm{sign}_2 x$ determined by the equalities

$$\mathrm{Sign}_1 x = \begin{cases} 1 \text{ at } x \geqslant 0 \\ 0 \text{ at } x < 0 \end{cases} \qquad \mathrm{Sign}_2 x = \begin{cases} 1 \text{ at } x > 0 \\ 0 \text{ at } x \leqslant 0 \end{cases}$$

3. The operation of adding a unit: $x+1$

4. The operation of addition $x+y$ modulo 2^{ℓ}

5. The operation of subtraction $x-y$ modulo 2^{ℓ}

6. The function $\pi(x,y,z)$ executing the counting by modulo y with resetting to initial x-value

$$\pi(x,y,z+1) = \begin{cases} \pi(x,y,z)+1 \text{ if } \pi(x,y,z) \neq y \\ x \qquad \text{if } \pi(x,y,z) = y \end{cases}$$

The function $\pi(x,y,z)$ may be expressed by the preceding functions, but it plays an important role in the synthesis of the program, and therefore we use for this function a special designation.

7. The operation of formation over any group of arguments $\varphi_p(x,y,z,\dots,\omega)$ at which the words representing the values $x,y,z,\dots,\omega$ are written from left to right, thus forming the resulting word.

33

We shall combine the functions enumerated above by substituting the arguments by functions. At the same time we shall admit the so-called "time transformation" according to the following rule. We assume that the set of recursive functions $f_n^1, f_n^2, \dots, f_n^s$ has been determined and another set of recursive functions has been formed from the variable $\mathcal{N}$:

$$\varphi_{\mathcal{N}}^1, \varphi_{\mathcal{N}}^2, \dots, \varphi_{\mathcal{N}}^t$$

Then the set of the functions,

$$f_{\varphi_{\mathcal{N}}^1}^1, f_{\varphi_{\mathcal{N}}^1}^2, \dots, f_{\varphi_{\mathcal{N}}^1}$$

we shall designate as the set obtained by "time transformation". This kind of procedure is very useful for building additional cycles of lower degree into the program.

Recursive functions originating from this class allow to record conveniently a number of computing algorithms. As an example may be cited the recording of an algorithm for the solution of a set of linear equations

$$x_i = \sum_{k=1} a_{ik} x_k + b_i \qquad (3)$$

by the Seidel iteration method.

$$x_i^{(s+1)} = \sum_{k=1}^{i-1} a_{ik} x_k^{(s+1)} + \sum_{k=i}^{\mathcal{N}} a_{ik} x_k^{(s)} + b_i \qquad (4)$$

breaking off at the condition

$$\max \left| x_i^{(s+1)} - x_i^{(s)} \right| < \varepsilon \qquad (5)$$

$$\text{or } S = S_0$$

Let the addresses of the coefficients a_{ik} be $(i-1)\mathcal{N} + k - 1 + \xi_0$, the addresses of the values in S approximation $(x_i^{(s)}) = \xi_1 + i - 1$, and the addresses of the values in the $S+1$ approximation $(x_i^{(s+1)}) = \xi_2 + i - 1$. The addresses of the free members are chosen in form of

$$(b_i) = \xi_3 + i - 1$$

Now, the calculations according to the formula (4) are ensured by the commands

$$\left.\begin{array}{llll} (1) & y\psi(\mathcal{N}-1) + k - 1 + \xi_0 & \xi_8 + k - 1 & \beta_0, \\ (2) & C\alpha\beta_0 & \beta_1 & \xi_2 + i - 1, \end{array}\right\} (6)$$

where $\delta = 2$, if $i < K$ and $\delta = 1$ if $i \geq K$

The verification of the conditions (5) is made by means of the commands

$$\begin{array}{lllll} (1) & B\tau_2 & \xi_1 + i - 1 & \xi_2 + i - 1 & \beta_2 \\ (2) & B\tau_1 & \beta_2 & (\varepsilon) & \xi_1 + i - 1 \end{array} \qquad (7)$$

and by checking whether all the values $[\xi_1 + i - 1]$ are negative. Here $B\tau_1$ means ordinary subtraction, and $B\tau_2$ the formation of the difference absolute value.

First, let us obtain the recursive functions determining the addresses of the commands in conventional time. The calculations according to the commands (7) shall be made as the values in $S+1$ approximation are computed according to the commands (6).

We introduce the cells $\gamma_1, \gamma_2, \gamma_3, \gamma_0$ for three variable addresses in the commands (6) and the cells $\gamma_4, \gamma_5, \gamma_6$, storing the number of the factor K in the sum (4), the number of the line i, and the number of the executed S iteration respectively. The addresses in (7) are stored in the cells γ_0 and γ_*. The contents of these cells are determined as a set of recursive functions of the following form:

$$\left.\begin{array}{l}
[\gamma_4]_n = \pi(1, N, n), \\
[\gamma_5]_{n+1} = \{[\gamma_5]_n + S([\gamma_4]_n, N)\}(1 - S([\gamma_5]_n, N)) + S([\gamma_5]_n, N), \\
[\gamma_5]_0 = 1 \\
[\gamma_6]_{n+1} = [\gamma_6]_n + S([\gamma_5]_n, N), \\
[\gamma_6]_0 = 0. \\
[\gamma_1]_n = \pi(\xi_0, \xi_0 + N^2, n), \\
[\gamma_2]_n = \pi(\xi_1, \xi_1 + N, n), \\
[\gamma_3]_n = \pi(\xi_2, \xi_2 + N, n), \\
[\gamma_0]_0 = \xi_2 \\
[\gamma_0]_{n+1} = ([\gamma_0]_n + S(N, [\gamma_4]_n))(1 - S([\gamma_5]_n, N)) + S([\gamma_5]_n, N)\xi_2, \\
[\gamma_*]_0 = \xi_1, \\
[\gamma_*]_{n+1} = ([\gamma_*]_n + S(N, [\gamma_4]_n))(1 - S([\gamma_5]_n, N)) + S([\gamma_5]_n, N)\xi_1
\end{array}\right\} (8)$$

We introduce the function $[\varphi]_n$ which takes the value I in case the iterations are completed, and zero, in the opposite case. From the conditions in which the iteration process breaks up may be seen that:

$$[\varphi]_n=\Big\{1-\prod_{i=1}^{N}\big(1-\mathrm{sgn}[\xi_1+i-1]_n\big)\Big\}\,\mathrm{sgn}(S_0-[\gamma_6]_n)+1-\mathrm{sgn}(S_0-[\gamma_6]_n) \tag{9}$$

Now, for the command executed at the moment ν we can record

$$\begin{aligned}[\alpha_0]_\nu=\Big\{&S([\alpha_1]_\nu,1)\big\{\varphi_p(y[\gamma_1]_n,[\gamma_2]_n,\beta_0)\,\mathrm{sgn}_1([\gamma_5]_n-[\gamma_4]_n)+\\
&+\varphi_p(y,[\gamma_1]_n,[\gamma_3]_n,\beta_0)\,\mathrm{sgn}_2([\gamma_4]_n-[\gamma_5]_n)\big\}+\\
&+S([\alpha_1]_\nu,2)\,\varphi_p(C\ell,\beta_0,\beta_1,[\gamma_0]_n)+\\
&+S([\alpha_1]_\nu,3)\,\varphi_p(B\tau_2,[\gamma_*]_n,[\gamma_0]_n,\beta_2)+\\
&+S([\alpha_1]_\nu,4)\,\varphi_p(B\tau_1,\beta_2,(\varepsilon),[\gamma_*]_n)\Big\}(1-[\varphi]_n)+[\varphi_n]\,Cm.\end{aligned} \tag{10}$$

It remains to determine the function $[\alpha_1]_\nu$ and effect the "time transformation". The function $[\alpha_1]_\nu$ is determined by the conditions

$$\begin{aligned}&[\alpha_1]_0=1\\
&[\alpha_1]_{\nu+1}=\big\{([\alpha_1]_\nu+1)S([\alpha_1]_\nu,1)+S([\alpha_1]_\nu,2)\big\}(1-S([\gamma_4]_n,N))+\\
&\qquad+S([\gamma_4]_n,N)\big\{S([\alpha_1]_\nu,4)+([\alpha_1]_\nu+1)(1-S([\alpha_1]_\nu,4))\big\}\end{aligned} \tag{11}$$

Finally, the "time transformation" is determined by the function determined by the conditions $n=\Psi_\nu$, $\Psi_0=0$

$$\Psi_{\nu+1}=\Psi_\nu+(1-S([\gamma_4]_n,N))S([\alpha_1]_n,2)+S([\gamma_4]_n,N)S([\alpha_1]_\nu,4) \tag{12}$$

The set of relations (8), (9), (10), (11), (12) fully presets the program of calculations according to the Seidel method. These considerations form a sound basis for the choice of the control device structure. On similar considerations may be based the choice of the memory device structure and, in particular, the choice of the system of a multistage memory and for the use of the memory with successive read out.

25

Para.3. AUTOMATA AND PROGRAM OPERATORS

Let us consider the finite automat A, the state of which is characterized by the states " n " of its components e_i $(i=1,2,\dots,n)$. Let us assume that $\{n\}$ is the set of all integers from I to n. Let us consider the states at a certain moment of time, as an aggregate of values of a certain function f on a set of $\{n\}$; Thus it can be said that A at the given moment is in the state f. Since we are speaking of a discrete A, we may assume a certain " τ " which is the minimum necessary for the transition of A from one distinguishable state f_1 into the other state f_2. Let us designate τ - as the A step. We shall call by the term "command" the different operations that A is capable of executing during τ. Let A_0 be at a certain moment in the state $f_{(0)}$. As a result of the step performed, A has passed from the state $f_{(0)}$ into the state $f_{(1)}$. The operator K_1 is such that $f_{(1)}=K_1 f_{(0)}$ is the command operator /7/.

Suppose, that for each e_i there is some zero state. The state A will be zero (0) if all the e_i are in zero states. The zero-operator is such that $0f=0$. The unit operator E, determined from $Ef=f$ is the idle step operator.

If in A pass consecutively steps with operators $K_1, K_2,\dots,K_j,\dots$, the A will consecutively be in the states

$$f_{(1)}=K_1 f_{(0)}, f_{(2)}=K_2 K_1 f_{(0)}, \dots, f_{(j)}=K_j K_{j-1}\cdots K_1 f_{(0)}$$

Usually the operation of the automat is determined by a certain finite number of commands (the program), which are reproduced consecutively (the cycle). If the operators of these programs are K_j $(j=0,1,\dots,m)$, then P - the program operator - is equal to

$$P=K_m K_{m-1}\cdots K_2 K_1 \qquad (I).$$

Let F_0 be the starting state. As a result of the operation of A, shall be obtained the states $F_j=P^j F_0$ $(j=1,2,\dots,S,\dots)$ In such a manner operates a homogeneous automat, i.e. without any external modification of its states. There might be homogeneous automata, in which in the S cycle, after the operator K_{t_s} the state φ_s is exteriorly set, by means of the input operator G_s. Then, in the A are obtained the following states:

$$F_S=P_{t_s,m}\left[G_S(P_{1,t_s}F_{S-1},\varphi_S)\right], \qquad (S=1,2,\dots)$$

where $P_{t_s,m}=K_m\cdots K_{t_s+1}$, $P_{1,t_s}=K_{t_s}\cdots K_2 K_1$.

Two types of input operators are to be considered that is

1. When the introduced states are superposed on the states of A at the moment of input. Then

$$F_S = P_{t_S, m}[P_{1,t_S} F_{S-1} + \varphi_S]$$

2. When the introduced states replace the states of the A components at the moment of input. Then

$$F_S = P_{t_S, m}[P_{1,t_S} F_{S-1} + E_{\{\nu_S\}}(\varphi_S - P_{1,t_S} F_{S-1})]$$

where $\{\nu_S\}$ is the set by which φ_S differs from zero (i.e. the states of the elements e_i with numbers from $\{n\}-\{\nu_S\}$ remain unchanged), and $E_{\{\nu_S\}}$ is the projection of the operator E over the set $\{\nu_S\}$ ($E_{\{\nu_S\}} f = f$ over a set $\{\nu_S\}$ and $E_{\{\nu_S\}} f = 0$ in the points $\{n\}-\{\nu_S\}$).

The automat synthesis problem may beraised as follows: There are given - the operator U, the function $f_{(0)}$ and the commands aggregate $\{K\} = K_1, K_2, \ldots, K_j, \ldots, K_m$
On the basis of the operators from $\{K\}$ we must build-up an automat passing through the states $F_S = U^S f_{(0)}$
Let $\varphi_{(0)}$ be the starting state of A and P - its program operator and let A solve the problem. Then, presumably the following relations are to be executed

$$P^S \varphi_{(0)} = U^S f_{(0)} \qquad (S = 1, 2, \ldots)$$

Generally speaking, if U and $f_{(0)}$ are not in a certain way specialized, only $\varphi_{(0)} = f_{(0)}$ and $P = U$ are possible. This means, that at such a statement of the problem, the operator U must belong to the class of operators which are factorized into the product of command operators from $\{K\}$. However, the problem of synthesis acquires a greater interest in case when U either cannot at all be factorized into the product of operators from $\{K\}$ or it is factorized into the product obtained from a great number of such operators, which leads to the synthesis of the automat involving a great time of operation.

Therefore, it is expedient to bring to certain modification in the statement of the synthesis problem, and demand that the states of A coincide with the required states not for all components, but only for certain preset components. In other terms, the functions characterizing these states must coincide over a certain set $\{\nu\} \subset \{n\}$.

37

If we designate with $\mathcal{E}_{\{\nu\}}$ an operator which ensures that at any function " f " there will be $\mathcal{E}_{\{\nu\}} f = f$ over a set $\{\nu\}$ and $\mathcal{E}_{\{\nu\}} f = \xi$ over a set $\{n\}-\{\nu\}$ (ξ -is an arbitrary function), then for the synthesis P it shall suffice to realize the relation $P\mathcal{E}_{\{\nu\}} = \mathcal{E}_{\{\nu\}} U$ (2). Then, $\varphi_{(0)} = \mathcal{E}_{\{\nu\}} f_{(0)}$.

From this equation are to be determined P and $\mathcal{E}_{\{\nu\}}$ Let us consider now the statement of the synthesis problem for the case of a non-homogeneous automat. Additionally to the problem of the homogeneous automat synthesis for obtaining the required states, it is admitted that into each operation cycle of the automat is introduced from the outside of the state ψ .

For better certainty, let us assume that ψ is introduced at the end of the cycle. Let $\varphi_{(0)} = \mathcal{E}_{\{\nu\}} f_{(0)}$ and $\psi = R f_{(0)}$ For the input, according to the type 2, with the operator $E_{\{\gamma\}}$ it will suffice to realize the relations:

$$E_{\{n\}-\{\gamma\}} P\mathcal{E}_{\{\nu\}} - \mathcal{E}_{\{\nu\}} U = -E_{\{\gamma\}} R \tag{3}$$

at the condition superposed on the operator R - the condition of the invariability of the state ψ introduced from outside·

$$E_{\{\gamma\}}(RU - R) = 0 \tag{4}$$

for the input according to type I these relations take the following form:

$$P\mathcal{E}_{\{\nu\}} - \mathcal{E}_{\{\nu\}} U = -R \tag{5}$$

$$RU - R = 0 \tag{6}$$

These relations have determined a certain class of operators U for which may be synthesized the automat on the basis of operators $\{K_j\}$. Let us consider some possibilities of extending this class. Assume that T and M , operators of the type $\mathcal{E}_{\{\nu\}}$ -are commutating operators, with U and P respectively. In this case, for the synthesis of a homogeneous automat, with the adopted statement of the problem, it is sufficient to meet the relations

$$MP\mathcal{E}_{\{\nu\}} = \mathcal{E}_{\{\nu\}} TU$$

For the synthesis of a nonhomogeneous automat the respective relations shall be written as follows

$$M(P\mathcal{E}_{\{\nu\}} + R) = \mathcal{E}_{\{\nu\}} TU \tag{7}$$

$$MR - RTU = 0 \tag{8}$$

41

The operators M and T, as extending the possibilities of satisfying the conditions (2) - (6) for the synthesis of the automat may be called "automation multipliers" M - left-hand and T - right-hand multipliers. The multipliers T and M may be introduced in corresponding conditions both separately and together. The introduction of the multiplier T sets the following limitation on U :

U must belong to a class (let us name it the IId class) of operators which admits commutating operators of the type $\mathcal{E}\{\nu\}$. Of considerable interest from the point of view of automata synthesis is the introduction of left-hand multipliers. In this case, it is not obligatory for the operator U which is preset for realization to belong to the IId class, but this feature must necessarily possess the operator of the automat P . It seems expedient to differentiate the automata with operators belonging to the IId class. We may call such automats as high-quality automata - for they present greater facilities for meeting the synthesis conditions. The quality of such automata depends upon the arbitrariness in M . The greater the arbitrariness allowed in the determination of M for the given P the more are the possibilities to direct this arbitrariness for the satisfaction of the corresponding synthesis conditions.

We may discuss the question of satisfying the synthesis conditions in the sense of expanding the set $\{n\}$, over which is determined the preset operator U up to $\{n+m\}$. Here, is essential the conception of effective expansion of the operator U up to $\bar{U}$, i.e. such an expansion that $\mathcal{E}_{\{n\}}\bar{U}$ coincides with U . The problem of the automat synthesis with $\{n+m\}$ components for the realization of $\bar{U}$ is equivalent to the problem of the automat synthesis for the operator U. In this case into the automat are introduced auxiliary components, the operation of which aims at satisfying the synthesis conditions.

The purpose of the above considerations was to satisfy by means of various transformations the conditions of the synthesis, by automata, having program operators P which may be thoroughly studied. For the synthesis conditions it is not sufficient to know P , but it is necessary also to know the starting state $\varphi_{(0)}$. The solution of the equations (2), (3), (5) in relation to $\mathcal{E}_{\{\nu\}}$ allows to determine $\varphi_{(0)}=\mathcal{E}_{\{\nu\}}f_{(0)}$. However, it is a complicated procedure. A more simple method may be proposed. Let us determine the functions $f_{(i)}=U^{i}f_{(0)}$. Assume that

$$\{n\}=\{\nu\}+\{\nu_1\}+\cdots+\{\nu_s\}$$

where all the components, with the exception of $\{\nu_s\}$ are of equivalent power with $\{\nu\}$ and the power $\{\nu_s\}$ is less than the power $\{\nu\}$.

Moreover, all the components of this sum are assumed as non-crossing.

Let us designate by $\Pi_{\{\nu\},\{\nu_j\}}$ the commutating operator, which converts the ensemble $\{\nu\}$ into $\{\nu_j\}$ and in the case of $\{\nu_s\}$ converts into $\{\nu_j\}$ an equivalent part $\{\nu\}$.

Let us build up the operator $\tilde{P}$

$$\tilde{P} = E_{\{\nu\}} + E_{\{\nu_1\}} \Pi_{\{\nu\},\{\nu_1\}} E_{\{\nu\}} P + \cdots + E_{\{\nu_s\}} \Pi_{\{\nu\},\{\nu_s\}} E_{\{\nu_s\}} P^s$$

And the function $\tilde{f}$

$$\tilde{f} = E_{\{\nu\}} f_{(0)} + E_{\{\nu_1\}} \Pi_{\{\nu\},\{\nu_1\}} E_{\{\nu\}} f_1 + \cdots + E_{\{\nu_s\}} \Pi_{\{\nu\},\{\nu_s\}} E_{\{\nu\}} f_{(s)}$$

Then
$$\varphi_{(0)} = \tilde{P}^{-1} f$$

The synthesis problem may be stated for U preset in implicit form $F_{s-1} - UF_s = 0$
The transfer into an explicit form would require the determination of U^{-1} , which is of a complicated nature. It is possible to synthesize the automat for the operator U i.e., to determine $P = K_m \cdots K_1$ and then to pass over to $P^{-1} = K_1^{-1}, K_2^{-1}, \ldots, K_m^{-1}$, which solves the problem. Here we, naturally, assume that the inverse operators K_j^{-1} exist and are known, since the operators K_j are known. The starting function in this case is the same as in the synthesis of the automat for U .

In the above described statement of the synthesis, the automat reproduces the required states only in components, with numbers defined by the set $\{\nu\}$. While the states of other components, which were regarded as auxiliary at the synthesis of the automat for U , may have independent values at the synthesis of the automat for another function. In other words, the synthesized automat may correspond in the part of the components $\{\nu_1\}$ to the problem for the operator U_1 with an initial state $f_{(0)}$, while in the other part $\{\nu\}$ it will correspond to the problem for the operator U_2 with the initial state $f_{(0)}^2$. The relation between U_2 and U_1 in this case is determined by the equality:

$$\mathcal{E}_{\{\nu_1\}} U_1 \mathcal{E}^{-1}_{\{\nu_1\}} = \mathcal{E}_{\{\nu_2\}} U_2 \mathcal{E}^{-1}_{\{\nu_2\}}$$

The important particular case when $U_2 = U_1 = U$ may be realized, if $\mathcal{E}^{-1}_{\{\nu_2\}} \mathcal{E}_{\{\nu_1\}}$ may be commutated with U .

The questions discussed in this paper were directly related to the synthesis of automata. However, the same questions may be applied to the analysis of the automata. In particular may be determined the classes of operators, for the realization of which the given automat may be used. The problem of the operator P recovery may be also raised and in certain conditions solved on the basis the results of the operation of the automat if specially selected text functions are introduced into it.

x x x

Above are detailed different methods of approaching the logical description of the dynamics of computer functioning, these computers being regarded as finite automata of a special type. The method of time logical functions allows to represent the microstructure of the circuit as well as the functioning of registers and systems of registers.

The description of the automat in terms of time logical functions allows to characterize the number of the circuit components, the complexity of feedback systems, and the dynamics of the operation of separate components. The presentation of the same automat by means of a set of recursive functions allows to connect more tightly its logical structure with the operator scheme of the algorithm.

Of the greatest interest in this connection is the prospective of finding the most effective structure of the computer control system ensuring the efficient realization of different classes of algorithms. A more detailed study of the processes occurring at each operational step of the computing machine may be made by investigating the command operators, described in the present paper. In its turn, the method of recursive functions allows to preset the order of execution of these operators, and the method of time logical functions provides for the presetting of their circuit execution.

All the afore-considered methods of formalization of a logical description reflect different sides of a unique problem - the synthesis of an efficient structure of automata on the basis of a description of its structure.

At the same time the usual approach to the theory of automata is so to say phenomenological, i.e. it reflects the system of transitions between states, but is isolated (alienated) from concrete realization.

REFERENCES

1. Yu.Y. Basilevsky "On the theory of time logical functions". Collected papers: Questions related with the Theory of Mathematical Machines." Moscow, Fizmatizdat, 1958.

2. Yu.Y. Basilevsky "Solution of time equations by the reduction method." Report at the conference on the theory and uses of discrete automatic systems. Moscow, 1958.

3. Yu.Y. Basilevsky. "On some transformations of finite automata." Report at the conference on the theory and uses of discrete automatic systems. Moscow, 1958.

4. Yu.A. Schreider. "Programming and recursive functions. Transactions No.1.

5. A.A. Lyapounov. "On program logical schemes." Transactions "Problems of Cybernetics," I. Moscow. Fizmatizdat. 1958.

6. A.I. Ershov. "Operator algorithms and partial recursive functions. "DAN. USSR v.122. No.6 (1958).

7. I.Y. Akushsky. "On some general questions of programming." Transactions No.1.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

ANALYSIS OF THE WORKING PRINCIPLES OF SOME SELF ADJUSTING SYSTEMS IN ENGINEERING AND BIOLOGY

by

S.N. BRAINES, A.V. NAPALKOV,

Psychiatry Research Institute, Moscow, USSR.

and

Yu.A. SCHREIDER,

Electronic Mathematical Machines Research Institute,
Moscow, USSR.

The concept of control process arises in various branches of natural sciences and possesses a number of common features which are studied by cybernetics.

The most highly organized controlling processes are those related to higher nervous activity of animals and man. The study of these processes from the point of view of general cybernetics by revealing the laws of the corresponding control algorithms is of great interest. It gives an understanding of the nature of higher nervous activity, enriches the theory of control algorithms and helps to create new classes of effective algorithms. A considerable number of publications have been published by now, concerned with attempts to model various features of the control processes in the central nervous system. These include investigations on modelling conditional reflexes (1), (10), on the creation of various models of mice, turtles, etc. which imitate the behaviour of the live creatures (3), (6), (7), (8), works on the mathematical theory of learning (4), (9), etc.

It is more difficult to indicate publications which develop consistently a cybernetical approach to the study of actual questions of the physiology of higher nervous activity and to revealing the features of the corresponding control algorithms.

It is appropriate at this point to quote Academician I.P. Pavlov (2) who pointed out the necessity of employing mathematical analysis in studying processes of higher nervous activity.

This paper discusses questions of algorithm creation for complex control processes reflecting the features of formation of conditional reflex chains.

x x x

Para.1. GENERAL CONTROL SYSTEM

Any controlling process infers the existence of two devices - one controlling device and the other controlled - which exchange information. The complexity of the controlling device depends on the quantity of information it has to treat over a definite interval of time. The controlling device can employ the information about the controlled device in various ways. In the general case the controlling device may receive information as to the influence of the environment on the controlled device, characterized at each moment by a signal $\mathcal{F}(t)$, and information as to the result of the action of the controlled device, characterized by a signal $y(t)$. On the basis of this information the controlling device produces a signal $x(t)$ which acts on the controlled device. The algorithm of producing the signal $x(t)$ is determined by the condition that the signal $y(t)$ differs sufficiently little in some sense from a predetermined function $\tilde{y}(t)$. Thus the result of the action of the controlled device is a function of the controlling influence $x(\tau)$ and of the influences of the environment $\mathcal{F}(\tau)$ and $\psi(t)$, where $\tau \leq t$:

$$y(t) = \mathcal{D}(x(\tau), \mathcal{F}(\tau), \psi(t)) \qquad (1)$$

here $\psi(t)$ describes the part of the influence of the environment concerning which no information is received.

In its turn, the controlling influence is produced by the controlling device by realization of the algorithm A which determine $x(t)$ by the values of $\mathcal{F}(\tau)$ and $y(\tau)$ at $\tau < t$. The signals $\mathcal{F}(t)$ and $\psi(t)$, generally, are accidental. If $\mathcal{F}(t)$ and $\psi(t)$ are known functions the possibility of obtaining the required signal $\tilde{y}(t)$ at the output of the controlled device depends only on the permissible arbitrariness in selecting $x(t)$. In the contrary case, when the values of $\psi(t)$ are independent for various t and the right side of (1) depends substantially on the value of $\psi(t)$ the result $y(t)$ will inevitably possess a certain minimum degree of indefiniteness which cannot be lowered by virtue of the controlling influences. The most interesting case is the intermediate one when the values of $\mathcal{F}(t)$ and $\psi(t)$ are correlated at various moments of time.

In the following we assume that the time t runs through a discrete sequence of values (for instance all the integral values) and the indefiniteness of the signal is characterized by its entropy (see /5/).

In the case of the process of formation of a conditional reflex the above-described scheme of control can be integrated as follows.

The signal $\mathcal{F}(t)$ represents information on conditional stimuli; the signal $y(t)$ brings information on the reinforcements (unconditional stimuli), $x(t)$ is a characteristic of the occurrence of non-occurrence of the reaction (reflex), $\psi(t)$ characterizes the indefiniteness in the correspondence between the reinforcement and the stimuli.

The complexity of the controlling process is characterized by the total entropy H of information of the signals at various moments of time.

An important characteristic of the process is the value

$$H_\infty = \lim_{T \to \infty} \frac{1}{T} H\left(y(t), \mathcal{F}(\tau)\right)_{t-T<\tau<t} \tag{2}$$

It can be shown that the value H_∞ determines the minimum degree of indefiniteness (entropy) of the value $y(t)$ per unit time which can be obtained by an optimum method of control, i.e. by selection of the algorithm A.

Another important class of controlling systems are systems for which the following limit exists

$$\tilde{H} = \lim_{T \to \infty} \left[H\left(y(t), \mathcal{F}(\tau)\right) - TH_\infty\right]_{t-T<\tau<t} \tag{3}$$

The value $T=\tilde{T}$, at which the expression in brackets in (3) differs sufficiently little from its limit $\tilde{H}$, characterizes the time interval during which the information must be received to produce the controlling signal $x(t)$ ensuring a sufficiently good quality of control, i.e. entropy $H(y(t))$ close to $\tilde{H}$. The value $\tilde{H}$ itself characterizes the time necessary to produce the control algorithm A, i.e. the "learning time" $\tilde{t}$ according to the law

$$\tilde{t} \sim \tilde{T}\, 2^{\tilde{H}} \tag{4}$$

It is assumed that $\varphi(t)$ is a stationary stochastical process satisfying certain special limitations.

To study the dynamics of formation of chains of conditional reflexes it is important to consider the case where $\mathcal{F}(t)$ is a chain of successively acting conditional stimuli ($b_1, b_2, \ldots b_K$) $y(t)$ is a chain of successively acting reinforcing factors ($c_1, c_2, \ldots, c_K$), determined by the sequence of actions $x(t)$ ($a_1, a_2, \ldots a_K$).

We may consider a scheme in which each successive stimulus b_i is at the same time a reinforcing factor $b_i = c_{i-1}$ and the factor c_K is the final reinforcing factor (food, removal of pain).

This scheme decreases substantially. The next step in reducing $\tilde{H}$ is to establish the correlated subsequences of $\tilde{H}$ actions in combination with stimuli and to work out an algorithm from such pre-set subsequences of actions.

Experimental study of systems of conditional reflexes in animals enables disclosure of some algorithms of the work of the brain. Analysis of these algorithms is possible on the basis of the theoretical conceptions considered above. The schemes of these experiments and their qualitative interpretation is given in the following paragraphs.

//

Para.2. EXPERIMENTAL INVESTIGATION OF CONTROL ALGORITHMS IN FORMING CHAINS OF CONDITIONAL REFLEXES

The method used by us can be characterized as a version of the conditional reflex procedure developed by I.P. Pavlov (2). A valuable contribution to the study of this problem were the works of L.G. Voronin (12).

In the course of the experiment definite systems of outer regularities were created artificially in the experimental environment surrounding the animal. For instance, the experimenter would set the scheme:

$$\text{bell} \longrightarrow \boxed{\begin{array}{l}\text{jump to}\\ \text{pedestal No.1}\end{array}} \longrightarrow \text{whistle} \boxed{\begin{array}{l}\text{press}\\ \text{pedal}\end{array}} \longrightarrow \text{food}$$

According to this scheme the experimenter was to introduce various stimuli depending on the movements of the animal. In this way the experimenter played the part of the environment.

In the course of the experiment, by using various versions of the procedure studies could be made of the laws relating to the formation of a new working programme (chain of conditional reflexes) in the animal, enabling it to obtain food under these experimental conditions. The animal would ascertain the regularities of the environment created artificially by the experimenter (recorded preliminarily on paper) and unknown to it. On this basis it would develop its optimum working programme under the given conditions. This procedure, which was used by us in experiments with a human being as well, reveals the complex picture of interaction between the environment and the organism, observed in the process of developing new working programmes. It thus becomes possible to disclose a certain sequence of operations by means of which the new working programme can be developed. This sequence includes both a definite system in accomplishing moving reactions and definite principles in estimating and utilizing the information received from the environment.

On the whole, we can speak of a definite learning algorithm, i.e. of a definite sequence of operations by means of which a new working programme can be developed. It should be emphasized that in this case we do not mean an algorithm of simple behaviour of animals, but an algorithm of a special higher category, an algorithm of a higher order, through which animals can develop various new forms of algorithmic behaviour in any new situations of the environment.

The experimental procedure enables various outer situations to be created. These situations may differ both in the complexity of the regularities artificially created and in the probability of accidental (not depending on the actions of the animal) appearance of various stimuli. Along with rigid regularities accidental coincidences of stimuli may also be included in the outer situation.

Under these various conditions of the artificially created environment various algorithms could be revealed on which the formation of new working programmes are based.

The procedure employed also enabled the study of algorithms in cases where the animals had already received partial or complete information of the experimental situation.

Investigations, carried out according to the above described procedure established that the chain of reflexes could be developed on the basis of double reinforcement by the successive addition of new links. In working out each new link of the chain one of the conditional stimuli of the earlier developed chain links is employed as an immediate reinforcement. The entire system after completion was reinforced by food.

The algorithm described ensures the appearance of new working programmes (behaviour algorithms) of the animal under various new conditions. It should be emphasized that this algorithm involves estimation and selection from all the information received by the brain, of that part of the information which is needed to build an optimum working programme. One of the criteria of estimating information is the principle of recurrent coincidence of two signals discovered by I.P. Pavlov. This criterion is quite reliable, because recurring coincidence may serve as a proof of the fact that in our case the organism has to deal with real laws of the outer world and not with accidental coincidence of two stimuli. The criterion of usefulness of information is at first a temporary coincidence of this information with food and afterwards with a conditional signal, in its turn, previously related to food.

It is important to stress that in the course of development of a system of reflexes new "landmarks" keep arising which serve to estimate both the usefulness of various movements and the usefulness of the information received by the control system. These "landmarks" include all conditional stimuli of newly developed conditional reflexes. The appearance of new

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

intermediate "landmarks" may be of great importance for developing new chains of conditional reflexes (new working programmes). The importance of the appearance of these "landmarks" consists in artificial limitation of the entropy by decreasing the number of situations considered, i.e. by the eliminating the necessity of fully considering all the situations.

Para.3. MORE COMPLEX CONTROL ALGORITHMS

The case considered above was an extremely simple one. As is known, much more complex programmes have to be used in cybernetic systems. Very often a possibility is provided to switch the work of a system from one programme to another depending on the results of the work of the system.

There arises the question whether such complex programmes can be developed in self-organizing systems. This involves both the problem of finding the corresponding algorithms and the question of their realization in cybernetic machines. The experiments were made according to the described procedure, but the nature of the regularities of the environment set up in the course of the experiment were more complex. For instance, a stimulus was introduced making accomplishment of the reflex chain impossible. Or such a system of environment regularities would be set up by virtue of which two stimuli would have to be present simultaneously for any definite action to be accomplished.

The corresponding diagrams are:

$$\begin{array}{l} \quad\; a_4\ б_8\ a_8\ б_{10}\ a_2 \longrightarrow C\ \text{(food)} \\ a_3\ б_1\ a_4\ б_7 \longrightarrow б_6 \end{array} \qquad (5)$$

In this scheme the presence of $б_6$ made accomplishment of the main reflex system impossible. At the same time there was a potential possibility of finding a way of eliminating $б_6$ by means of the reflex chain $a_3\ б_1\ a_4\ б_7$ or

$$\left.\begin{array}{l} a_4\ б_8\ a_6\ б_{10} \\ a_3\ б_1\ a_4\ б_7 \end{array}\right\} a_2 \longrightarrow C\text{(food)} \qquad (6)$$

In this case to accomplish the action a_2 two signals $б_{10}$ and $б_7$ had to be present. Such systems of interrelations are often encountered. It was established that when animals were placed under such experimental conditions they worked out the corresponding system of reflexes. The laws of development of these systems of reflexes were studied. Particularly, it was established that exclusion of the stimulus $б_6$ (diagram (5)) and introduction of $б_7$ and $б_{10}$ (diagram (6)) may serve as a basis for the formation of a new chain of reflexes. Development of this chain of reflexes no longer required direct reinforcement by food. These chains thus differed from ordinary food reflex chains. They possessed a certain "autonomy". (As is known, the elements of a food chain disappear quickly if they are not reinforced with food).

On the basis of these experiments certain conclusions can be drawn as to the nature of the algorithms lying at the basis of the formation of complex systems of conditional reflexes. An important element of these algorithms is the appearance of a new trial of all possible movements in conditions of the presence of stimulus $\mathcal{B}_6$ (diagram (5)) and the absence of stimuli $\mathcal{B}_{10}$ and $\mathcal{B}_7$ (diagram (6)).

If, as a result of any movement, stimulus $\mathcal{B}_6$ disappeared or stimuli $\mathcal{B}_{10}$ or $\mathcal{B}_7$ appeared, a new "autonomous" chain of reflexes began to form, this chain being no longer related directly to food. In these experiments it was demonstrated that systems of conditional reflexes may have a very complex structure.

Para. 4. ALGORITHMS UTILIZING PREVIOUSLY DEVELOPED CHAINS OF CONDITIONAL REFLEXES

The most general case is the situation when the system possesses partial information about the controlled object. Evidently, in this case the algorithm of the work of the system should make it possible to draw precisely the information needed at that particular moment from the memory and to include it in a strictly definite part of the newly formed working programme.

This system must possess a memory and mechanisms for selection of the necessary information.

Let a certain control system have a definite number of previously developed working programmes. There is a certain concrete situation (a set of signals $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ etc. entering the system). Under these conditions the system is confronted by the task of developing a new programme of work by means of which it can accomplish a certain new result. It may also be required that certain dangerous states of the controlled object should be evaded.

The work of the controlled system must evidently result in a certain new programme which should correspond to the given concrete situation ($б_1, б_2, б_3$) and accomplish the same purpose. In other words a certain new expedient form of behaviour must be worked out.

This new programme arises on the basis of a retreatment of previously accumulated information. In the course of retreatment of all the information stored in the memory that part of it must be selected which may be of use in achieving the end and can be utilized in the given concrete situation represented by a definite set of signals ($б_1, б_2, б_3, \ldots$).

The information is stored in the form of a large number of work programmes of various kinds and developed at different occasions.

To analyze this scheme various kinds of complex conditional reflex systems reinforced by food, pain disappearance stimuli, etc. were developed in the animals (13).

In the course of the experiment some new demand was brought up. For instance, thirst was caused in dogs and a certain new set of external stimuli was started.

Under these conditions new forms of behaviour developed in the animal connected with the reception of water, including definite sections of previously developed food and defensive reflex chains. Experiments of this kind made it possible to study the algorithms of formation of new working programmes when partial or complete information on the controlled object is available. During this experiment the previously developed reflex systems and the experimentally created situation (set of signals) were known exactly, so that each part of the new working programme could be traced to the corresponding previously developed chain. This made it possible to observe the process of formation of the new programme.

By varying the form of the experiment certain essential laws could be established characterizing the formation of new reflex systems on the basis of newly accumulated information.

It was found that when any definite situations were created new systems of conditional reflex reactions arose immediately in the animals without additional development.

These working programmes corresponded each time exactly to the experimental environment created artificially and resulted in satisfaction of the new demand of the animal (reception of water).

These acts of behaviour could not be reduced to the sum total of stimuli included in the set. Each act of behaviour was a unified integral system and represented an integral reaction of the organism accomplished in response to the entire set of signals presented. The presence or absence of any one stimulus in the set entirely alters the newly arising system of conditional reflexes.

It was found that under ordinary conditions animals would not react at all to conditional stimuli, though the reflex system was stably developed. In creating thirst (excitement of the drinking centre) conditional reflex reactions were observed to appear in response to definite conditional stimuli. Here, activation of the individual reactions was of a selective nature. The order of distribution of this activity depended on the whole set of signals employed. The animals' reactions were always of an integral nature.

In studying this phenomenon definite laws were established. A set of special "starting" conditional irritators were detected which, themselves not causing any moving reaction, put into action a certain section of the system. A certain order of subordination was detected in the action of these signals. One of the **stimuli** A (a stimulus of the highest category) would put a definite sector of a previously developed reflex system into an active state. Other stimuli ($B_1, B_2, B_3, \ldots$) would start up individual parts of this sector, but their action could manifest itself only after stimulus A had been brought into play. A third group of stimuli ($C_1, C_2, C_3, \ldots$) would start up individual refle· chains. Their action was found to be possible only after A and B_1 had been put into action.

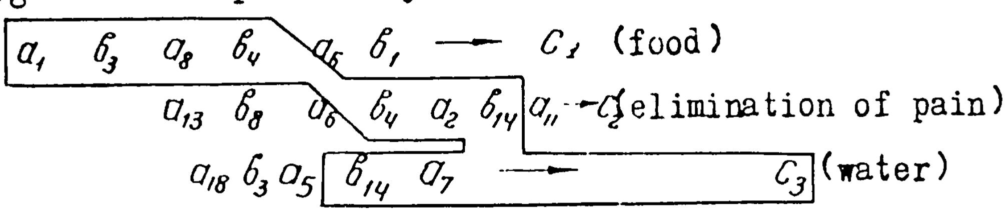
Then the following regularities were discovered. If thirst was caused, the first conditional reflexes to become active were those of the drinking chains. And the first of these conditional reflexes to go into action were those immediately connected with unconditional reinforcement (water). If any of these conditional stimuli were present in the experimentally created environment the other conditional reflexes would not become active. But if these stimuli were absent other conditional reflex reactions began to become active. If at least one common stimulus of two unlike reflex chains (the food and the drink systems) was present the animal would begin to respond to the conditional irritators of the food reflex system. If a common irritator of the food and the defence chains was present a certain section of the defence reflex chain also became active. If there were no signals common to the two unlike reflex chains no conditional reflex reactions of those systems were observed to become active.

/11

The above-described process of the consecutive rise of conditional reflex responses to a definite group of stimuli depended also, as was indicated above, on the presence in the environment of definite "starting" stimuli. This caused the phenomenon of subordination just described. Any definite chain of reflexes could become active only if the corresponding starting signals were present.

On the basis of modern data of neurology and cybernetics the following hypothesis can be put forth to explain the facts presented above. When thirst is caused the resulting excitation process begins to spread over systems of previously developed conditional reflex ties. This lowers the excitement threshold of the corresponding first elements. If, as the excitation spreads, another excitation caused by an external conditional stimulus arrives at the same nerve cells the two excitations add up and result in the corresponding conditional reflex movement. The necessary information is selected as a result of these processes.

The union of unlike reflex chains (food and drink, and defence, etc.) can be explained in a similar manner. If a common stimulus is present the excitation process may spread from one reflex system to another. Synthesis of the new working programme is explained by the following diagram:



It can be seen from this diagram that the algorithm of union of the systems by the principle of a common stimulus may result in the synthesis of a new system of reflexes $a_1\,б_3\,a_8\,б_4\,a_2\,б_{14}\,a_7 \longrightarrow c_3$ (water) serving for the procurement of water and consisting of various sections of previously developed systems.

The brain work algorithms connected with the utilization of previously accumulated information which we have studied provide for rapid formation of new forms of behaviour.

Analysis of the principles of formation of conditional reflex chains contributes to progress in the study of questions of the physiology of a higher nervous activity in that it helps to establish the structure of the algorithms controlling the complex behaviour of higher animals and offers an opportunity of finding ways of economically realizing controlling algorithms in cybernetics systems.

REFERENCES

1. A.A. Lyapunov. Basic Problems of Cybernetics. Collection "Problems of Cybernetics" Moscow, 1958.

2. I.P. Pavlov. Twenty Years' Experience in the Study of Higher Nervous Activity (Behaviour) of Animals. Collection of Articles, Leningrad, 1938.

3. I.A. Poletayev. Signal, Sovetskoe Radio, 1958.

4. R. Bush. F. Mosteller, Statistical Theory of Learning, 1956. The Mathematical Theory.

5. C.E. Shannon, W. Weaver. The Mathematical Theory of Communication, Urbana, 1949.

6. V.G. Walter. Imitation of Life. Scient. Amer. 1950. Vol. 182, No.5.

7. E.S. Berily. A Seeing Electronic Animal. Radio Electronics, 1951, Vol.23, No.3.

8. E. Eichler. Artificial Turtle. Radio Technik (Radio-amateur), 1955, Vol.31, No.516.

9. U. Ross Ashby. Diagram of an amplifier of thinking ability. Collection of Articles, Edited by N.E. Shannon and J. McCarthy.

10. O.M. Attlee. Conditional Probability and Conditional Reflex Machines. ibid.

11. N. Wiener. Cybernetics. New York, 1948.

12. L.G. Voronin. Analysis and Synthesis of Complex Stimuli in Higher Animals, Moscow, 1953.

13. S.P. Braines and A.V. Napalkov. Directional Irradiation of the Excitation Process According to Systems of Previously Developed Ties. 18th Conference on Problems of Higher Nervous Activity (Abstracts of Reports, Leningrad, 1958).

12

METHODS OF SPEEDING-UP THE OPERATION OF DIGITAL COMPUTERS

I.Y. AKUSHSKY, L.B. EMELIANOV-YAROSLAVSKY,
E.I. KLYAMKO, V.S. LINSKY, G.D. MONAKHOV,

Institute for Scientific Research of Electronic Mathematical Machines. Moscow, USSR.

METHODS OF SPEEDING-UP THE OPERATION OF DIGITAL COMPUTERS

I.Y. AKUSHSKY, L.B. EMELIANOV-YAROSLAVSKY,
E.I. KLYAMKO, V.S. LINSKY, G.D. MONAKHOV,

Institute for Scientific Research of Electronic Mathematical Machines. Moscow, USSR.

INTRODUCTION

All the various methods of accelerating the execution of operations, considered as one of the means for speeding-up calculations, may be characterized by one common feature, that is, the applicability of such methods does not depend of the concrete contents of the program.

Speeding-up calculations is achieved by the accelerated execution of the elementary "bricks" of the program, i.e. the computing operations.

The present paper does not deal with the methods of acceleration based on definite program characteristics (for example, the selection of the command system, address, memory organization and the use of assembled computing systems).

The methods of acceleration of computing operations may be classified into two groups by the nature of these operations, that is:

1. Logical methods of speeding-up the main computing operations.

2. Methods of accelerated calculation of elementary functions.

Para.1. ON THE PRINCIPLES OF ACCELERATION OF THE EXECUTION OF OPERATIONS

Any computing operation executed by the machine may be dismembered into a certain sequence of simple actions. Let us designate every such simple action, effected at the entry of some control signal from the control device, by the term EMO- elementary machine operation. Though this concept is not quite precise, it may be useful in many cases.

Thus, a computing operation may be regarded as an aggregate of various elementary machine operations performed in a certain order.

Obviously, this aggregate is not determined only by the computing operation to be executed, for the list of elementary machine operations depends, as well, upon the adopted algorithm. However, for the majority of basic computing operations a long time since was developed and at present definitely accepted a "classic" system of used elementary machine operations. To these belong elementary operations of binary addition, shift, code transfer and so on.

The same may be said about the methods of dismembering the computing operations into sequences of components of elementary machine operations. Here, too, there are firmly established rules and recommendations.

However, the classic algorithms of the execution of computing operations are not the most efficient from the point of view of rapidity of machine actions and, in cases of especially high speed operation requirements, they can not be considered as optimum. Evidently, this conclusion being quite trustworthy, may be made beforehand, otherwise, would be very little verisimilar the following statements:

a) the set of classical elementary machine operations is the optimum;

b) any sequence of elementary machine operations at the execution of a computing operation is algorithmically the shortest, i.e. it can not be replaced by a sequence with a smaller number of members. It may be possible to set and solve the problem of determining all accelerated algorithms, proceeding from a sufficiently large class of possible elementary machine operations, but being formulated in this manner the problem would result extremely complex and practically hardly feasible to solve it.

Efficient algorithms may be determined much easier by an artificial way.

Indicated below, are some accelerated algorithms obtained in this manner.

From the previous analysis can be made a classification of the algorithmetic methods for the accelerated execution of computing operations.

1. The method of reducing the number of consecutively executed elementary machine operations by:

a) elimination of superfluous elementary machine operations;

b) simultaneous execution of elementary machine operations.

2. The method of introducing special elementary machine operations (special in the sense that they differ from classic operations),

There is a third method of speeding-up the execution of computing operations which differs essentially from the algorithmetic methods, i.e.:

3. The method of reducing the time of elementary machine operation by selecting an efficient logical structure of employed circuits.

Para.2. ALGORITHMETIC METHODS OF SPEEDING-UP OPERATIONS

Let us consider first the algorithms of accelerated execution of multiplication operations.

According to known statistical data about 50% of the machine time is spent on the execution of multiplication operations. Therefore, speeding up this operation is of great importance.

The difference between various methods of accelerated execution of multiplication operations lies in their machine algorithms.

There are electronic digital computers in which the multiplication is performed as a single elementary machine operation. However, more frequently the machine algorithm of the multiplication operation takes the form of alternating shifts over one digit and additions (digit by digit multiplication). The first method of multiplication gives satisfactory results as regards speed of operation, but requires a great number of equipment (in quadratic relation to the number of digits).

At the digit by digit multiplication may be used a combined (coincidence-type) adder as well as a counter-type adder.

Combined adders involve the use of a serial (or serial-parallel) digit transfer.

However, the stored sum of partial products may be easier formed in a counter-type adder. Therefore, we shall deal with machine algorithms of the multiplication operation employing adders of this kind.

Speeding-up by overlapping of elementary machine operations of addition and shift, is achieved in the simplest case by shifting the multiplicand code at the moment of the following addition in the adder.

The sum of partial products at multiplication, according to this method, remains unmoved.

It is interesting to note, that with this method of multiplication, if no doubled accuracy products are required ("2 n" digits), the adder and multiplicand register may be executed with " n " digits. For this, a ring shift of the multiplicand code should be made in the multiplicand register, and the adder should have a circuit of cycle carry from the highest to the lowest order. Moreover, prior to each transfer of the shifted multiplicand code on the adder, in the latter are to be cleared the memory cells corresponding to the highest order of the transfered code.

Wide use has found the multiplying circuit in which the shift of the partial product is effected in the adder during the addition EMO and does not involve additional operations of components.

An interesting alternative procedure of speeding-up of the multiplication by overlapping is the method of the "travelling wave". According to this method, in the process of multiplication, the addition of several partial products is accomplished simultaneously in the same adder. This method involves the use of a special counter-type adder, in which a new addition may be started from the lowest orders side before the previous addition (or even several precedent additions) in the higher orders has been completed.

This method of multiplication permits to store a sum of partial products at a maximum frequency, conditioned by the admissible frequency of component functioning.

The methods of speeding-up the multiplication operation by way of reducing the number of executed addition and shift EMO, are based on the elimination of addition steps at the multiplication by the multiplier digits, in the code "0" and on the elimination of the addition steps at the multiplication by multiplier digit groups in the code "I".

The reduction of the number of shift steps in the two aforementioned cases is effected by the introduction of elementary machine operations of group shifts over 2,4,8 digits and so on, or over an arbitrary number of digits designated by "K".

In case the digits of the multiplier y contain many "I" the multiplication may be made using the formula $x \cdot y = \overline{x \cdot \overline{y}} + x$ where the sign $\overline{}$ designates the EMO of conversion into the reverse code. Time is gained in this case because the code $\overline{y}$ contains a small number of "I".

Maximum speeding-up of the operation is achieved by the introduction into the arithmetic device of a special arrangement for shifting the multiplicand code over an arbitrary number of digits "K", during one shift step (see further) and by a special conversion of the multiplier code S', defining the minimum number of additions and subtractions necessary in the process of multiplication, according to the scheme.

$$S'\left[0{,}I00IIII0III\right] = 0{,}I0I000\text{-}I00\text{-}I \; .$$

The average number of EMO of addition - subtraction in this case, as shown by appropriate calculations, is equal to:

$$\frac{1}{8} \cdot \left[\frac{80 + 24\,(n-2)}{9}\right] - \frac{1}{2} + (-1)^{n} \cdot \frac{1}{18} \;;$$

where n - is the number of digits.

It may be easily shown that by employing only classical EMO, a considerable speeding-up of operations can not be obtained. Indeed, any multiplication algorithm in this case is defined by a special identical conversion of S' digits of the multiplier y. Obviously, $SS' = S'$ if S' corresponds to the most efficient algorithm. Let us assume that Sy and $S'y$ do not coincide and consider the case when the higher digit

31

Sy is greater than the higher digit $S'y$. (The case when S and S' interchange their roles is also considered analogously). Then,

$$Sy - S'y = Sy - SS'y \geqslant 0{,}10\text{-}10\text{-}10\ldots\ldots - 0{,}010101\ldots > 0$$

which is impossible.

For the multiplication operation may be used the method of "carry remembering", first proposed by M. Nadler (Czechoslovakia), which involves a special EMO. This method takes advantage of a specific feature of the multiplication operation consisting in the fact that, at this operation all intermediate actions, preparing the result, are executed by the circuit without "conditional" transfers. Owing to this, it is possible to considerably reduce the time of the multiplication operation by introducing a special addition EMO with incompleted carry. The carries occurring at addition during this EMO are memorized in a special register.

The memorized carries are taken in account at every new addition and cleared at the end of the operation. A considerable acceleration of the multiplication operation may also be obtained, if the memorizing of the carries is effected not in every digit, but in several equally distanced points of the adder.

Let us now consider the algorithms of accelerated division. Usually, the quotient digit is defined by the feature of the direct or reverse (additional) code of the remainder. However, the remainder code, besides the aforementioned information, contains some additional data, which frequently allows to determine at once the group of quotient digits and, thus, reduce the number of elementary machine operations. The idea of this method is that when a remainder is formed with a sufficiently small or sufficiently big absolute value, the following digits of the quotient shall be obligatorily a group of identical digits (zeros or units).

Let us assume that the divisor q is normalized, i.e. contains "1" in the highest digit. Obviously, if the code of the positive remainder contains in its highest digits a "K" number of zeros, then, besides "1", in the quotient digits are to be recorded also K - 1 zeros. For obtaining the next remainder, it is sufficient to simply shift the initial remainder to the left over "K" digits and subtract the divisor from the obtained number.

The case with the negative remainder A_0 of a sufficiently small absolute value is symmetrical to the just considered. Assume, that in the higher digits of the remainder code there are K units. Demonstrate, that K-1 following remainders will be positive and, consequently, besides "0" in the next quotient digits should be recorded K-1 units.

Evidently, the i^{th} remainder of A_i is equal to $A_i = 2^i A_0 + q$ provided all previous remainders were positive.

From the study of the A_0 code, and taking into account the normalization of the divisor q, it may be concluded, that when $i \leq K-1$ all remainders A_i are positive, i.e., as required.

The last A_K remainder may be obtained by shifting the A_0 code over K digits to the left and adding the number thus obtained to the divisor q.

Let now A_0 be a positive remainder approaching closely enough to q (such cases are more rarely encountered). This fact may be found out in the machine by means of a simple circuit analyzing the higher digits of the divisor and remainder.

In this case it is necessary to build up the quantity $A_0' = A_0 - q$ and record in the quotient the K units contained in the higher digits of this quantity. The next remainder is obtained by shifting the A_0' code to the left over K digits and adding the divisor q.

The case when a negative remainder A_0 of a big absolute value is obtained, is symmetrical to the case just considered.

The average number of M quotient digits, obtained in one addition or subtraction, EMO (taking into account only small values of remainders) is determined for the case of numbers with many digits in the following way:

$$M = \frac{1}{2} + \frac{1}{4} + \sum_{K=3}^{\infty} \frac{K-1}{2^K} = \frac{3}{2}.$$

It may be demonstrated, that the indicated division algorithm (taking into account only small remainders) is the most effective cycle algorithm, which realizes the method digit by digit, on condition that besides the divider register is used only one adding register and only classic EMO.

Let us assume that in the i^{th} cycle in the adder is obtained a certain quantity $A(r_{i-1}, q) = r_i$ defining the following group of digits of the quotient $B(r_i)$. (A and B are designations of certain algorithms). It may be easily ascertained that all the r_i are nothing else but remainders and, consequently, $B(r_i)$ is the maximum determinable group of the quotient digits. However, minding that of the divisor q i, only known that it is normalized, the method for obtaining the digit group described before, exhausts all the possibilities; as was necessary to demonstrate.

The machine algorithm of the division operation with simultaneous shift of the divisor is usually not employed for the reason that it either results in a loss of division signs and less accuracy in division, or requires a divisor register and adder of double length.

However, by using a ring shift of the divisor it is possible to eliminate the effect of shifts on the duration of the division operation without increasing the equipment and without any loss in accuracy.

This method presupposes the utilization of reverse codes and the presence of a circuit of cycle carry in the adder. Apparently, the shift of the remainder code on the adder may be practically not made, considering that the place of the point is "moved" over 1 digit to the right.

Correspondingly, on the adder is to be transferred the divisor with a ring shift to the right. At every such transfer, the position of the point and, correspondingly the position of the sign digit are "shifted" in the adder over 1 digit to the right. The further actions are obvious.

As an example of the execution of the division operation with the utilization of special EMO, may be cited the method of M. Nadler, realized by means of the addition EMO with an incompleted carry. However, in some cases, it is even possible that the sign of the remainder, i.e., of the quotient digit will be determined incorrectly. If it is assumed that in each digit of the quotient there is also "-1", then, using this method, any error in any one of the digits may be corrected at the expense of the following digits.

The operation is completed by reducing the obtained quotient to the ordinary form by subtracting two codes, corresponding to the positive and negative units in the quotient digits.

11

Speeding-up of operations may be achieved in many cases by eliminating normalization and presenting the numbers in not normalized condition. Thus, without considerable loss in accuracy, it is possible at multiplication to have only one of the multipliers normalized, and, moreover, its normalization may be partially overlapped in time with memory access of the other multiplier. If the result of addition is to participate in the subsequent addition, its normalization to the left is also not obligatory.

A certain speeding-up may be achieved by the representation of negative numbers in the machine in the reverse code, on condition that besides the sign is introduced the code feature. In this case, at algebraic addition, no time is wasted in the adder for the conversion - it coincides with the transfer of the code into the adder (at multiplication - into the multiplicand or multiplier register). It is expedient to introduce the code feature for digits as well.

It seems expedient to increase in the machine the number of active computing devices, capable of conducting computing operations in parallel and separately from each other, and capable to ensure a wide direct exchange of information by interaction. The presence of several active computing devices permits to obtain a more effective execution of comflexes of operations, as well as separate arithmetic operations. Let us consider the possible procedures for the realization of certain operations in conditions of an increased number of components of the arithmetic device.

Assume that:

$\{T\}_i^{\tau}$ - is the condition of the device (adder-S, register -R) at the τ-cycle of the i-stage of the process.

$\{T\}^{\to p}$, $\{T\}^{\leftarrow p}$ -shifts, to the left and to the right respectively, over p digits.

1. Calculation of $u = ab^x$ with multiplication in a descending order of degrees 2^x.

If

$$b_j = \varepsilon_n 2^j + \varepsilon_{n-1} 2^{j-1} + \dots + \varepsilon_{n-j}\,; \qquad b = b_n \tag{1}$$

Then: $b_{k+1} = 2b_k + \varepsilon_{n-k-1}$ u $b_{k+1}^2 = 2^2 b_k^2 + \varepsilon_{n-k-1} 2^2 b_k + \varepsilon_{n-k-1}$

Take part S_b, S_{b^2}, R_a u R_b

α) $\varepsilon_{n-k-1} = 0$

I) $\{S_b\}^1_{n-k-1} = \{S_b\}^{\leftarrow 1}_{n-k-2}$; $\{S_{b^2}\}^1_{n-k-1} = \{S_{b^2}\}^{\leftarrow 2}_{n-k-2}$

β) $\varepsilon_{n-k-1} = 1$

I) $\{S_b\}^1_{n-k-1} = \{S_b\}_{n-k-2} + \{R_a\}^{\to 2}_{n-k-3}$; $\{S_{b^2}\}^1_{n-k-1} = \{S_{b^2}\}^{\leftarrow 1}_{n-k-2}$

2) $\{S_b\}^2_{n-k-1} = \{S_b\}^{1\leftarrow 1}_{n-k-1}$; $\{S_{b^2}\}^2_{n-k-1} = \{S_{b^2}\}^1_{n-k-1} + \{S_b\}^1_{n-k-1}$

3) $\{S_b\}^3_{n-k-1} = \{S_b\}^{2\leftarrow 1}_{n-k-1}$; $\{S_{b^2}\}^3_{n-k-1} = \{S_{b^2}\}^2_{n-k-1} + \{R_a\}^{\to 1}_{n-k-2}$

B) Calculation of $u = ab^2$ with multiplication in an ascending order of degrees 2^x.

Let us designate: $\tilde{b}_j = \varepsilon_j 2^j + \varepsilon_{j-1} 2^{j-1} + \dots + \varepsilon_0$; $\tilde{b}_n = b$

$\bar{b}_j = 2^{j+2} \tilde{b}_j$

Then: $\bar{b}_{k+1} = 2\bar{b}_k + \varepsilon_{k+1} 2^{2(k+2)}$ u $\tilde{b}^2_{k+1} = \tilde{b}^2_k + \varepsilon_{k+1} \bar{b}_k + \varepsilon_{k+1} 2^{2(k+1)}$

α) $\varepsilon_j = 0$

I) $\{S_b\}^1_j = \{S_b\}^{\leftarrow 1}_{j-1}$; $\{R_a\}^1_j = \{R_a\}^{\leftarrow 2}_{j-1}$

β) $\varepsilon_j = 1$

I) $\{S_b\}^1_j = \{S_b\}^{\leftarrow 1}_{j-1}$; $\{S_{b^2}\}^1_j = \{S_{b^2}\}_{j-1} + \{S_b\}_{j-1}$

2) $\{S_b\}^2_j = \{S_b\}^1_j + \{R_a\}_{j-1}$; $\{S_{b^2}\}^2_j = \{S_{b^2}\}^1_j + \{R_a\}^{\to 2}_{j-1}$; $\{R_a\}_j = \{R_a\}^{\leftarrow 2}_{j-1}$

The schemes A and B require at each stage the knowledge of only one figure "b". Therefore they may be adapted for any process in which "b" is determined digit by digit (for

7-

example $b=\frac{D}{d}$, then $q=a\left(\frac{D}{d}\right)^2$).

The schemes considered above, may be realized in decimal code arithmetic devices, provided certain modifications, are brought in the schemes for the reason that doubling may be effected in the decimal code adder. Thus, multiplication of "a" by "b" in the descending order of degrees of 2^x is based on the presentation

$$ab = \{[(a\cdot\varepsilon_n 2 + a\,\varepsilon_{n-1})2 + a\,\varepsilon_{n-2}]2 + \cdots + a\,\varepsilon_0$$

and is realized in the machine arithmetic device from S_p and R_a by the operation $\{S_p\}_j^1 = \{S_p\}_{j-1} + \varepsilon_j\{R_a\}$

and $\{S_p\}_j^2 = \{S_p\}^{1\leftarrow 1} = \{S_p\}_j^1 + \{S_p\}_j^1$ and in the ascending order in the machine arithmetic device from S_p and S_a by the operation

$$\{S_p\}_j = \{S_p\}_{j-1} + \varepsilon_j\{S_a\}_{j-1}, \qquad \{S_a\}_j = \{S_a\}_{j-1}^{\leftarrow 1}$$

The direct execution of this multiplication requires the determination of binary digits. In this case it is advisable to use the well known decomposition of a proper decimal fraction into a binary fraction by the overflow of the adder S at the operation $S^{\leftarrow 1}$, obtaining binary digits in the descending order of the degrees 2^x. Minding the greater efficiency of multiplication in the ascending order, it is expedient to use the number b', dual in relation to "b" i.e.

$$b' = \varepsilon_0 2^n + \varepsilon_1 2^{n-1} + \cdots + \varepsilon_n$$

At division, binary digits of the quotient were obtained as a result of a corresponding trial and error procedure, and proceeding to the multiplication with these numbers, it is possible to form a quotient decimal code in the adder S_q (instead of R_q).

In the same manner may be modified the afore described schemes for execution in the decimal arithmetic devices.

Para.3. METHODS FOR REDUCING THE TIME OF ELEMENTARY MACHINE OPERATIONS

An important condition for the accelerated execution of the addition EMO is that each component of the add circuit is of single-shot type.

/ / / .

Usually in add circuits with single-shot operation components, for the addition procedure are used memory registers of both addends.

As a result it is possible to eliminate one of the memory operations, connected with the number transfer, as practiced in computing impulse adders. When it is not desirable to utilize in the add circuit the memory register of the 2-d number, the single-shot operation may be achieved by means of a scheme in which the code, of the number stored in the adder, is defined in each digit by the position of two memory components (the combinations "00" and "11" correspond to the code "0", while the combinations "10" and "01" correspond to the code "1").

The functions from the digit and from the carry in this circuit are divided between different memory components.

No simple mechanical solution has yet been found for reducing the time of the carries.

Adding devices in which the average time of the addition EMD is reduced by strictly noting the moment when the carry is completed, or by introducing "by-pass circuits" in the through carry circuit, are, for the time being, of extremely complicated design.

The problem of group shift acceleration may be solved by means of a special shifter.

The shifter (see Fig.1) is a ferrite matrix in which the information is simultaneously recorded on all the ferrites of the given column, (each matrix column corresponds to a certain digit of the recorded information).

All the ferrites of each matrix line are transpierced by a common reading wire. Besides the recording and reading wires, all ferrites which enter into one matrix diagonal, are transpierced by common shift wires.

In this manner, when a reading signal is applied to a bus, the number shifted over a certain number of digits in the direct or reverse code is read out.

The number of apparatus in this shifter is not greater than in usual shifting registers, but its functional diagram is more simple.

Para. 4. CIRCUIT EXECUTION ALGORITHMS FOR COMPUTING ELEMENTARY FUNCTIONS

The calculation of elementary functions should be included in the list of main machine operations adaptable for circuit execution may be considered the algorithms for the calculation of function values, developed from the "digit by digit" algorithms in the following directions.

Direct Scheme. a) Specially selected trial and error codes are periodically generated by transmitters;

b) The result of the trial and error procedure defines the method of formation of quantities by the arithmetic device. These quantities are consecutive approximations to the value $f(x)$.

Reverse Scheme. The trial and error codes are consecutively formed by the machine arithmetic device;

b) The value $f(x)$ is formed on the basis of the trial and error results from specially selected codes which are periodically generated by the code transmitter.

As elementary machine operations for circuit execution of the calculation of the values $f(x)$ may be adopted:

a) Addition $a_i \pm \varepsilon_i b_i$,

b) addition with shift $a_i \pm \varepsilon_i b_i 2^{z_i}$

The last elementary machine operation may be frequently used at $a_i = b_i$

$$b') \quad a_i \pm \varepsilon_i a_i 2^{z_i} = a_i (1 \pm \varepsilon_i 2^{z_i})$$

Elementary machine operation

b' - multiplication by the numbers $(1 \pm \varepsilon_i 2^{z_i})$ $\varepsilon_i = 0,1$

depending on the results of the trial and error procedure.

It should be noted that any number z may be represented with an accuracy up to 2^{-}_{p} in the interval $(1,2) - z = \prod_{0}^{p} (1+\varepsilon_i 2^{-i})$ in the interval $(0,1) - z = \prod_{0} (1-\varepsilon_i 2^{-i})$

///

Let us consider the execution of certain elementary functions on the basis of elementary machine operations a, b, b'

For the function $e^{(\pm)x}$ we start from the presentation

$$y = e^{(\pm)x} = (1 \pm \varepsilon_1 2^{-1})(1 \pm \varepsilon_2 2^{-2}) \dots (1 \pm \varepsilon_p 2^{-p}) \qquad (1)$$

The trial and error codes have the following form:

$$a_i = \ln (1 \pm 2^{-i}) \qquad (i = 1, 2, \dots, p)$$

Using the recurrent relation

$$y_{j+1} = y_j \pm \varepsilon_{j+1} 2^{-(j+1)} y_j \qquad (2)$$
$$y_0 = 1$$

we obtain the circuit of a device comprising an adder, binary code registers and a shifter (see Fig.3).

Depending upon the results of the consecutive trial and error, we calculate y_{j+1} by the addition of y_j to y_j shifted over $j+1$ digit to the right. At $y_0 = C$ we obtain the values of function, $C e^{(\pm)x}$.

By means of the reverse scheme, we can calculate the function $y = \ln x$.

Using the periodically generated numbers (2) and the recurrent relation

$$x_{j+1} = x_j + \varepsilon_{j+1} 2^{-(j+1)} x_j$$

we may determine ε_{j+1}

$$\varepsilon_{j+1} = \operatorname{Sgn} (x - \bar{x}_{j+1})$$

where $\bar{x}_{j+1} = x_j + 2^{-(j+1)} x_j$.

Depending upon the defined ε_{j+1} it is determined whether the addend a_{j+1} participates or not in the formation of the quantity $\ln x$.

For calculating the values of the functions $y = tg\, x$ it is expedient to adopt the following presentation

$$\tilde{x}_j = \varepsilon_1 x_1 + \varepsilon_2 x_2 + \dots + \varepsilon_j x_j \qquad (j = 1, 2, \dots, p)$$
$$x = \tilde{x}_p$$

$$tg\, \tilde{x}_{j+1} = \frac{A_{j+1}}{B_{j+1}}$$
$$x_i = arc\, tg\, 2^{-i}$$

11.

In this case, for the values A_{j+1} and B_{j+1} the following recurrent relations valuable take place:

$$A_{j+1} = A_j + \varepsilon_{j+1} 2^{(j+1)} B_j$$
$$B_{j+1} = B_j - \varepsilon_{j+1} 2^{-(j+1)} A_j \qquad (j = 1, 2, \ldots, p) \qquad (3)$$
$$A_0 = 0; \quad B_0 = 1$$

(with $A_0 = A_1$, $B_0 = B$ we obtain $\frac{A_p}{B_p} = \frac{A + B\,tg x}{B - A\,tg x}$

The values x_i are trial and error codes, on the basis of which are determined the ε_{j+1}

$$\varepsilon_{j+1} = Sgn\,(x - \tilde{x}_j - x_{j+1})$$

Next, by adding the shift (at $\varepsilon_{j+1} = 1$) are calculated A_{j+1} and B_{j+1} (or at $\varepsilon_{j+1} = 0$, $A_{j+1} = A_j$, $B_{j+1} = B_j$) and so on up to $j = p$. For determining $tg x$ it is necessary to make the division $\frac{A_p}{B_p}$.

For calculating the values of hyperbolic functions may be applied a procedure similar to that described for the exponential function.

The calculation of the values of the function $y = arctg x$ is made by solving the equation

$$\delta = A - B\,tg\,y = 0$$

The trial and error procedure is effected for determining ε_{j+1} by the sign of the value δ_{j+1} where $\delta_{j+1} = A_{j+1} - \bar{B}_{j+1}$

The values A_{j+1} and B_{j+1} are determined by (3) and $\bar{A}_{j+1}$ and $\bar{B}_{j+1}$ from relations similar to (3).

$$\bar{A}_{j+1} = \bar{A}_j + 2^{-(j+1)} \bar{B}_j$$
$$\bar{B}_{j+1} = \bar{B}_j - 2^{-(j+1)} \bar{A}_j$$
$$\bar{A}_0 = 0 \quad \bar{B}_0 = tg\,y$$

For carrying out calculations in decimal arithmetic devices, the before mentioned algorithms, must be so modified that any shift to the right is excluded.

For the function $y = e^x$ the recurrent relation (2) is replaced by

$$\bar{y}_{j+1} = 2^{j+1}\,\bar{y}_j + \varepsilon_{j+1}\,\bar{y}_j \qquad (2)$$
$$\bar{y}_0 = 2^{-\frac{p(p+1)}{2}}$$

/ / /.

where $\bar{y}_p = y_p = e^x$

For calculating the values $y = \ln x$ it is necessary to use the recurrent relation similar to (2') i.e.

$$\tilde{x}_{j+1} = 2^{j+1}\tilde{x}_j + \varepsilon_{j+1}\tilde{x}_j$$
$$\tilde{x}_0 = 2^{-\frac{p(p+1)}{2}}$$

Then
$$\bar{\bar{x}}_{j+1} = 2^{j+1}\tilde{x}_j + \tilde{x}_j$$

For detection of ε_{j+1} it is necessary to calculate

$$\operatorname{Sgn}(x - \bar{x}_{j+1}) = \operatorname{Sgn}(2^{\frac{(j+1)(j+2)}{2}} x - \tilde{x}_j)$$

When calculating the values of the function $y = \operatorname{tg} x$ the relations (3) are replaced by

$$A_{j+1} = 2^{j+1} A_j + \varepsilon_{j+1} B_j \qquad (3')$$
$$B_{j+1} = 2^{j+1} B_j - \varepsilon_{j+1} A_j \qquad (j = 1, 2, \dots, p).$$
$$A_0 = 0, \qquad B_0 = 1$$

Para.5. SPEEDING-UP OPERATIONS AT MICROPROGRAM CONTROL

The use of microprogram control in digital computers, presents many positive features and also speeds-up machine operation.

This is achieved by introducing in the external alphabet of the machine several symbols of accelerated algorithms, included in the executed program as a characteristic elementary link.

For other terms, this is achieved by forming new computing operations which are characteristic for the executed program and allows to make the most efficient use of the equipment. Such methods of speeding-up calculations are intermediate between the methods considered before and methods related with the concrete pecularities of the program.

Thus, the gain in time, in this case, is the difference between the time needed for executing the calculations according to standard programs composed of external alphabet

/ / /

operations for a usual machines and the time for executing the microprograms in the machine with microcontrol. This difference is mainly caused by the following:

1. Microprograms do not include such operations as control transfer and memory of certain command addresses for matching the main program with subprograms.

2. It is not necessary for many algorithms to bring intermediate results to a standard form, for example, rounding off and normalization may be omitted.

3. It is possible to match in time different operations, as for example, simultaneously with the addition performed in the arithmetic device adder it is sometimes possible to read out from the memory the digits for the following actions.

4. In some cases the specific features of the algorithm may be used to advantage.

As known, the reverse value $\frac{1}{x}$, may be computed by the iterative formula

$$y_{i+1} = y_i(2 - x y_i); \quad (i = 0, 1, \dots, i_k) \quad y_{i_k} \approx \frac{1}{x}$$

Let us replace x by $2\tilde{x}$:

$$y_{i+1} = 2 y_i (1 - \tilde{x} y_i)$$

In this form the formula becomes convenient in that all the numbers participating in the calculation according to this formula are not more than a unit, provided a limited interval of x modification is used and the initial approximation y_0 has been appropriately chosen. (The multiplication by 2 may be performed by a shift or addition of the number with itself). This allows to make the calculations with a fixed point. Moreover, no time has to be spent on the subtraction, as $1 - \tilde{x} y_i$ represents an additional code $\tilde{x} y_i$, which at all iterations may be replaced by a reverse code without loss in precision.

It is known that at calculations according to the indicated iteration formula the number of true signs y_i is doubled with each iteration. Thus, in y_i may be left $n \cdot 2^{-i_k + i}$ of higher digits and all others may be discarded. This may be used to advantage for reducing the time spent on multiplication by employing y_i as multipliers. The average number of additions in multiplications in this case will be equal to n. The initial approximation y_0 may be computed by the formulas of the type:

$$y_0 = \alpha x + \beta$$
$$y_0 = \alpha + (\beta - x)^2$$

where α and β are constants. In this case the time of multiplication may also be reduced at the account of the small number of digits. It may be considered that as far as speed is concerned, the method just considered may compete with different methods of division digit by digit, including the accelerated methods, as well. At present, when single-sided high-speed memories of large capacity on paper lists, have appeared, it has become possible, by increasing the number of constants, to pass from one approximation polynomial for the elementary function to a series of such polynomials at separate intervals.

Let us assume that the function value is to be determined at the interval $0 \leq x \leq 1$. We divide this interval in two equal ones by their length, and, in each of them, make an approximation of the function by the polynomial in the s^{th} degree. For each x the group $s+1$ of constants will be determined according to the q of the highest digits of the number . The lowest $n-q$ digits represent the difference Δx between x and the nearest lowest table value of the argument. The approximating polynomials are calculated by the Horner diagram with s multiplications

$$(\ldots(\alpha_s \Delta x + \alpha_{s-1})\Delta x + \alpha_{s-2})\ldots + \alpha_0$$

Obviously, in the polynomials, represented in this way, the coefficients $\alpha_1, \alpha_2, \ldots, \alpha_s$ as may be expressed without an exact number of digits. As the values $(\Delta x)^i$ after the point, have not less than q_i zeros, i.e., the number of significant digits which they contain is not more than $n-q$ consequently, the coefficients α_i may have not more than $n-q_i$ significant digits. The advantage of this method of polynomial representation appears at multiplication, owing to the fact that these values may be taken with an uncomplete number of significant digits.

It may be easily seen that the first multiplier contains on the average $\frac{1}{2}(n-sq)$ units, the second $\frac{1}{2}[n-q(s-1)]$ units and the last $\frac{1}{2}(n-q)$ units. Thanks to this, the calculation of the polynomial is considerably accelerated.

The analysis of the methods for calculating the elementary functions $\frac{1}{x}, \sqrt{x}, e^x, \ln x, \sin x, \operatorname{tg} x, \arcsin x, \operatorname{arctg} x$ shows that provided slight modifications and additions are brought in the usual arithmetic device it may be used for the execution of these methods. Thus, for computing $\sqrt{x}$ it is necessary to provide an output in the control device of the last digit of the order, and the possibility of high-speed recording and adding constants in the order adder.

Under these conditions, the speed of calculation of elementary functions in the arithmetic device controlled by a microprogram of an appropriate arrangement is increased by several times.

Thus, the control by microprogram ensures the possibility of making efficient use of all executive organs contained in the machine.

REFERENCES

I. R.K. Rychards. Arithmetic operations of digital computers. Foreign literature publishing house, I957.

2. V.S. Linsky. Computing elementary functions on digital computers. Computing mathematics. Issue 2. The Publishing of the Academy of Sciences, I958.

3. G.D. Monakhov and E.I. Klyamko. Method of Speeding binary division in digital computers. Priborostroenje N 2, I957.

4. I.Y. Akushsky. Many-registered schemes for executing arithmetical operations. The theory of mathenatical machines. Collection I, Moscow I958.

17

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

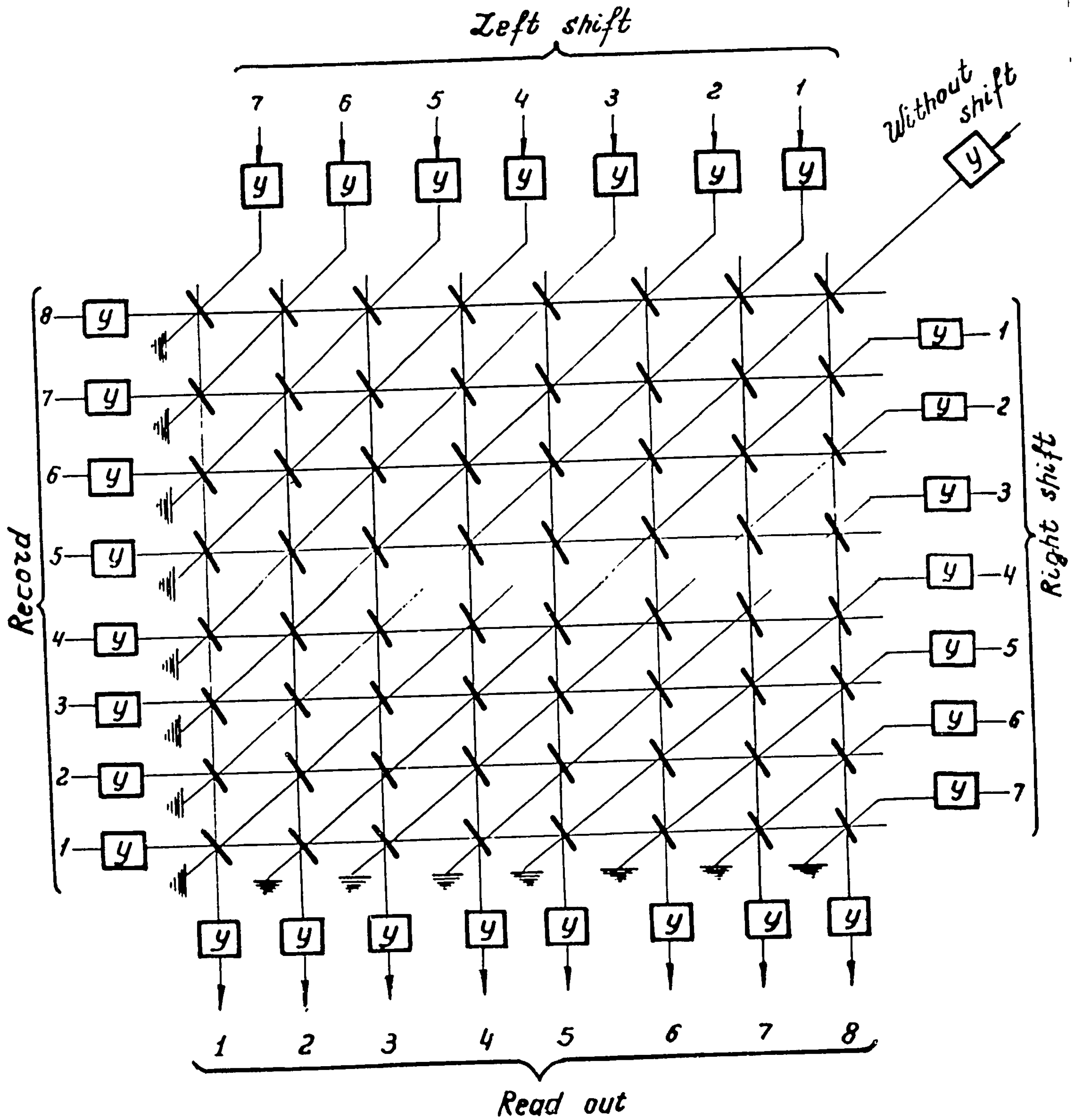


Fig. N1. Shifter.

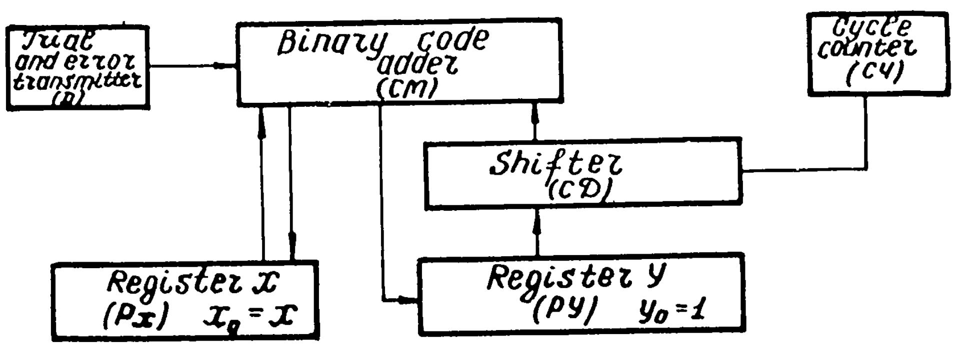


Fig. №2

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

Poor Original

MACHINE TRANSLATION METHODS AND THEIR APPLICATION TO ANGLO-RUSSIAN SCHEME

I.K.Belskaya,

Academy of Sciences of the USSR

In this paper an account is given of a scientific research which has resulted in devising an algorithmic procedure for machine translation of different languages into Russian /I/. Methods evolved for translational purposes are explained, the Anglo Russian scheme being chosen as an illustration of their application.

I. INTRODUCTION

Research in MT methods, which are outlined below, was started late in 1954 on the initiative of Academician A.S.Nesmejanov, President of the USSR Academy of Sciences. The first experiments in MT from English into Russian were carried out in December, [illegible] /I, 2/, which terminated the first stage of the research.

Some of the principles on which our research is based were put forward in earlier publications, among [illegible]

Poor Original

shed in RESEARCH, October 1957/3/, can also be mentioned.

Since then considerable progress has been made towards adequate formulation of the method. We are now in the position to say, that the second stage of the research has recently been completed, in the course of which the suggested methods were shown to be of general applicability, for which purpose these methods were extended to cover MT from languages differing from English in structure as much as Japanese, Russian, Chinese and German /4/.

As to the Anglo-Russian scheme of MT the research here has reached a stage where complete grammatical analysis at a bilingual level as well as rearrangement of most important types of English idiomatic constructions can be accomplished, grammatical modification of the Russian translation (which indeed is the simpler part of the problem) being performed by an independent set of routines, termed Russian Synthesis.

In addition to this, the progress in Anglo-Russian MT has taken the form of considerable growth of the volume of words now entered into the MT dictionary. More than 2000 words are stored in the English section of our multilingual MT dictionary, a still greater number of Russian equivalents being stored in its Russian section. The dictionary thus is made to cover different fields of applied mathematics I/

To complete this stage of research a large-scale test of the Anglo-Russian sheme has been carried out. IOO samples (which amounted to

I/ Participants in this work were G.A.Tarasova, whose contribution with compilation of the Anglo-Russian Dictionary is most valuable, and L.M.Bykova

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

amounted to 3000 sentences) of 'unknown text' were selected at random from different English authors, and translated into Russian in strict accordance with instructions provided by the MT dictionary any transl·tional routines[1/]. The ten persons chosen to carry out the experiment had no knowledge of English nor had they any previous experience with the tasks required[2/].

It emerged from the text that the scheme is very effective at dealing with all sorts of texts restricted, lexically, to applied mathematics, whereas grammatically no limitation as to type of the written text has been found necessary. 1 or 2 words per printed page is the average for 'unknown' words with the present size dictionary, which makes the translation quite adequate for iunderstanding (See Tables Nos 1,2,3,4).

For this reason as well as for reasons of preserving the proposed series of MT dictionaries strictly specialized as to field, we are not inclined to increase the volume of words in the present dictionary, but rather proceed with compiling medium size (say, 2500-3000 words each) dictionaries for various fields. This indeed will be our occupation at the next stage of research.

Translational routines for Anglo-Russian MT being final achievement of the recent research, it seems very reasonable, in the present communication, to lay particular stress on discription of translational routines for vocabulary and grammatical analysis of the English sentence. As to the principles on which MT vocabulary

---

1/ A.I.Martynova was engaged as superviser in the testing procedure.

2/ Several samples translated in this manner are given in Tables Nos 1,2,3 and 4.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

Poor Original

is based, the reader is referred to our earlier publications /3/.

2. General considerations. Applicability of MT methods.

Of two most general MT problems - those of possibility of machine translation and of its applicability - the former has already been resolved, both theoretically and practically, whereas the latter problem still remains open for discussion. The objective of the present research is to prove applicability of MT methods to any sphere of language.

To date, it is only within the limited sphere of scientific writing that the applicability of MT methods has won general recognition. As to other uses of MT most machine translators are inclined to feel very doubtful /4/.

However, the majority of restrictions imposed on MT application, when analyzed, turn out to be due to a very strong inclination on the part of investigators to describe the translated language /sourse language/ in terms of correspondence to some other system, say, another language, or a group of languages, or science other than linguistics, especially logics or particular fields in mathematics. The possibilities of MT are discussed then as dependent on common elements in the compared systems. These elements may be more or less numerous, yet absence of complete correspondence between the systems, which is usually the case, inevitably brings about limitations to the scope of MT. Thus, application of machines to translating literary works of art has more than once been declared as absolutely ruled out (See, for instance, Ref. 4, p.42)

In our opinion, it seems very reasonable to expect that these limitations can easily be eliminated, should the problem be formulated in a different way, namely, 'whether it is possible, within any language existing, to give formal description to any of its multiple spheres, individual as they may seem?'

This comes to the same thing as saying that the applicability of MT depends on whether it is possible to identify the implicit set of rules governing this of that particular sphere of language applications, be it as narrow a sphere as, say, Wordsworth's poetry, and, further, whether these rules can be formulated into a formal set.

Apparently, every piece of writing (insofar as written language is discussed) can be analyzed on these lines within the sphere where it belongs, and a set of rules for such analysis can be laid out. It is essential that these rules should be formal all along the line. Yet this is no obstacle either, since language is but a formal system of specific character developed by man to give communicative experession to his mental activities. As a consequence of the foregoing, it is immediately obvious, that problems posed by stylistic peculiarities of literary works of art can satisfactorily by resolved, if treated on the lines suggested above, i.e. within the sphere where they belong.

In this light, the supposed 'principal informalizability' of poetry (See Ref.4) should be rejected. Contrary to this supposition, poetry, as indeed any piece of literary art where formal elements are of no minor importance, is particularly susceptible to machine translation, in this sense.

This assumption has partly been justified on empirical grounds, that is, by experimental translation of passages from Ch.Dickens I/

I/ Illustration to be found in Reference /3/.

J. Galsworthy, J. Aldridge and Edgar A.Poe. /See Tables Nos 2, 3 and 4/. It is our firm belief, that further investigations will completely eliminate the restrictions imposed now on MT application.

An adequate description of a language, as indeed of any particular sphere of it, should finally aim at establishing within the analyzed system a set of correlations of the following type - $\boxed{\text{means} \rightleftarrows \text{effect}}$, by which the correlation of linguistic means and their meaning (effect) is inderstood.

Taken in its most general sense, the translational problem is, in effect, the problem of equating the aforesaid correlations of one language with those of another.

The procedure can symbolically be expressed by the following chart (See Fig.I.)

$$\boxed{\text{mea}\quad \rightleftarrows \text{effect}_I}$$
$$\|$$
$$\boxed{\text{effect}_2 \rightleftarrows \text{means}_2},$$

Fig.I.

where 'effect$_I$' and 'effect$_2$' are identical whereas 'means$_I$' and 'means$_2$' differ.

In the course of this substitution of one language for another, transposition of semantic content from one language to another is realized.

In conclusion, a final word must be added on the problem of MT prerequisites. These do not rest upon the existence of common basic elements in languages, as is often pointed out, but rather include the following two factors:

I. language in itself is but a system of formal means by which communication of meaning is effected;

Poor Original

2. all language systems existing are so developed as to express in their particular ways any shade of meaning as well as various emotional effects.

When falling back on our symbolization, this comes to saying that the number of 'effects' in any two languages is equal, which makes the corresponding systems of 'means' fully comparable, through their 'effects'.

Since language systems are formal, any application of them can be provided with a description programmable on a machine.

3. A Short Outline of Translational Routines.

General procedure covered by translational routines can be broken down into three independent steps, these being:

I. Vocabulary Analysis of the source language for which purpose MT dictionary and a set of dictionary routines are used;

II. Grammatical Analysis of the source language for which purpose Analysis routines are devised;

III. Grammatical Synthesis of the target language for which ends the same set of Synthesis routines is applied to the texts translated from different source languages.

To make the outline concrete, the translational routines will further be described in their Anglo-Russian realization I/

A. Dictionary Analysis.

Dictionary Analysis of the English sentence starts with searching every word of the text in the MT dictionary. The first dictionary

I/ Complete list of translational routines given in the order of their application to be found in Table No 3.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

Poor Original

routine to be used here is that of transforming words of the text into the standard forms listed in the MT dictionary (See Table No 5).

Thus 'wanted' will be transformed into 'want', 'stopped' into 'stop', 'coming' into 'come', 'lying' into 'lie', 'copies' into 'copy', 'bigger' into 'big', etc.

When dictionary search is completed, another routine is applied which concerns itself with the words that for various reasons have not been found in the dictionary. These are termed, 'unknown words', because their lexical equivalents remain unknown throughout the translational procedure. Yet, for the forthcoming grammatical Analysis, it is essential that grammatical qualification of the 'unknown words' should be obtained.

It is impossible to foresee every word in every text of a language or even of its particular sphere, since some of them may be occurring for the first time in the language, not to mention quite a number of more trivial reasons.

However, the 'unknown words' do not affect the translation, so far as grammatically they have been classified. To meet the latter problem a very important routine, that of classifying 'unknown words' into 'parts of speech' has been devised, where extensive use is made of morphology and syntax of these words.

Another category of sentence constituents which undergo preliminary grammatical analysis in accordance with a dictionary routine, are the so-called 'formulas', by which various symbols used in different sciences are understood. Syntactical function of every 'formula' in the sentence is defined in accordance with a special routine.

Poor Original

So much for the words and symbols not found in MT dictionary. In addition to lexical equivalents, words found in the dictionary are provided with information (termed 'invariant charactetistics') which is partly grammatical partly semantic in character. For more detailed discription of this information the reader is referred to our earlier publication /3/. The only thing that need be mentioned here, is that within the 'invariant characteristics' obtained from the dictionary final and preliminary information is distinguished. Information is considered final for dictionary cycle when lexical equivalent of the word is included. Instead, preliminary information of the word is restricted to the indication 'homonymous' or 'polysemantic'.

Special routines have been devised to deal with homonymous and polysemantic words, the analysis of former words preceding that of the latter.

The four types of 'Homonyms' analyzed by the routine, are those of 'adjective-noun' (Homonym I), 'noun-verb' (Homonym 2), 'verb-adjective' (Homonym 3) and of 'preposition-adverb' (Homonym 4). Among Homonyms I, a more complicated sub-type is distinguished, that of 'adjective-noun-verb' (Homonym I Complicated). See fig.2:

Cp.:	CHECK:	I. adjective	- КОНТРОЛЬНЫЙ
		2. noun	- КОНТРОЛЬ
		3. verb	- 'polysemantic'.
	SQUARE:	I. adjective	- КВАДРАТНЫЙ
		2. noun	- 'polysemantic';
		3. verb	- ВОЗВЕСТИ В КВАДРАТ

Fig.2.

In identifying Homonyms I, 2 and 3 a combination of morphological and syntactical analysis of the word is used. Thus, any inflection (except for ER or EST) identified in Homonyms I or 3 makes 'adjective' an impossible alternative, just as ED or ING inflexion in Homonym 2 cross out 'noun' solution. These are morphological criteria, which do not, however, find as wide an application as syntactical analysis does in view of scarce inflections in English.

The information Homonyms acquired in the course of this analysis may or may not be final for the dictionary cycle, since some of them are provided here with the indication 'polysemantic' instead of lexical equivalent (See Fig.2.)

Total number of polysemantic words stored in our dictionary amounts to 500 words.

Determination of multiple meaning is performed by specifying typical contexts of polysemantic words in acordance with a special routine which concludes Dictionary Analysis of the English sentence.

Since basic principles of this routine have been discussed elsewhere (See Ref.3), we do not propose to dwell on them here. However, to make this paper comprehensible on its own, two illustrations of context analysis of polysemantic words have been included (See Tables Nos. 6 and 7). For details and the background of the method, the reader is referred to our earlier publication /3/.

B. Grammatical Analysis.

Grammatical processing of a sentence is broken up into two inde-

[illegible] Analysis and Synthesis. The latter is the simplest of the two, for which reason it is not here that the main interest of the problem lies. So far, the discussion of Synthesis will be restricted to a few general comments.

Synthesis routines provide rules for grammatical modification of the translated text in accordance with grammatical information obtained in the course of the Analysis of the English original.

The most important peculiarity of Synthesis routines is their non-comparative nature, which means that rules of word-changing, as well as certain rules of word-building, are formulated strictly within particular target language. Owing to this, the same Synthesis routines can be applied to sentences translated from different languages.

However, Synthesis requirements are inclined to increase in case these routines serve multi-lingual MT purposes. With multi-lingual MT in view, Synthesis routines should be:

1. EXHAUSTIVE in describing target-language word-changing system, since grammatical rules with no application in MT from one language may become vitally important when the source language is changed;
2. IMPELLING in carrying through <u>any</u> instruction obtained from the Analysis of the source language, which makes necessary providing every 'non-productive' category of the target language with a 'productive' grammatical equivalent.

So far, the problem of grammatical equivalents within a language, theoretically, stands out as most important for Synthesis in MT.

3. Synthesis routines should be INDEPENDENT of Analysis, since the

Poor Original

... for different languages.

... 'Independent' Analysis cannot be recommended, ... would not at all help to make it economical. Analysis problems being numerous and important scientifically, they indeed deserve special discussion which is given below.

English Analysis is covered by six routines, which are applied in the order indicated in Table No 8. In view of length limitations of the present communication the discussion is restricted to general outline of most important Analysis routines, among which 'Verb' and 'Syntax' stand out as routines playing the key part in the whole procedure of Analysis.

8.1 The 'Verb Analysis' routine is divided into five sections, the first section being compulsory for every verb of the sentence, whereas of remaining sections only one is employed for each type of the analyzed verbs.

In Section I verb selection for further analysis is performed. Among words picked out for analysis in this routine are those possessing the indication 'Verb', so far as they do not have any of the following indications: 'to be Disregarded (D)' 'Not to be Changed' (NtbC), or (Russian) indications 'Participle', 'Verbal Adverb' or 'Verbal Noun'. Check-up for absence of these indications is meant to exclude from further analysis those of the verbs that have been elsewhere provided with characteristics that satisfies synthesis routines.

In addition to verb selection, correction of certain verb indications is envisaged in Section I.

Among verb indications liable to correction are those of ten-

... with verb... in if-clauses and of c se overnment with ... verbs, as well as some more particular indications. Analysis of homonymous forms, such as Past Indefinite and Subjunctive, of irregular verbs also belongs here.

...eck-up for grammatical context implying correction as possibility is performed both when one of the above-mentioned indications is ascribed to the analyzed verb in the dictionary and when it is about to be developed in the course of further Analysis.

...reliminary check-u..s of ection I are followed by verb analysis ..roper, for which purpose the analyzed verb is sent to one of the four different sections, differences in morphological structure of the verb being decisive in choosing the section. Thus, verbs with ..-ending1/ are sent to Section II, verbs with ED-ending, as well as certain forms of irregular verbs, enter Section III, verbs with ..G-ending are directed to Section IV, whereas verbs not inflected are analyzed in Section V.

...matical qualification of **'S-verbs' in Section II** depends ... or stand out as the only ending of the verb, or another ...ndi.. (usually ...) is associated with it. In the latter case, the following indications are developed for the Russian equivalent verb: 'Verbal Noun; euter; Plural', which imply further analysis by the 'Noun' routine at the proper time.

1/ The term ENDING is applied to affixes following the stem of the word, whereas affixes preceding the stem are called PREFIXES.

N o t e, that the term AFFIX is restricted to those formatives that are used in word-changing, whereas, formatives used for derivation are termed SUFFIXES.

[illegible] to the [illegible] ending, English characteristics of the verb
(predicate in the Present Indefinite form) is transformed into Russ-
[illegible] indications, but not without checking-up for correction con-
ditions (See above). Resultant characteristics is 'Predicate',
associated with either 'Present' or 'Future tense'. Number and
Person (or gender for the Past tense in other cases) of the Rus-
sian predicate remain not defined until the subject of the Russian
sentence is determined.

The analysis of 'ED-verbs', i.e. verbs with ED-ending and certain groups of irregular verbs, is performed in Section III, where syntax definitely takes precedence. The four main patterns of grammatical verb context analyzed here are indicated in Table No 9 as Patterns 1 : 1a, 1b, 1c, 1d;

2 : 2a;

3 : 3a, 3b;

4 : 4a, 4b, 4c.

Noteworthy is the fact, that contex analysis of a word implies, in all cases, observation of 'Rules of Word Selection'. These rules are based on classifying all the words in a sentence into three categories, which are:

I. words of third-degree structural significance, where particles, adverbials, parentheses and coordinated[1/] parts of the sentence; are included;

II. words of second-degree structural significance, where different words and word groups belong, so far as they are placed in the attributive position towards some word of a sentence;

III. words of first-degree structural significance, which include words not identified as belonging to either of the two pre-

1/ The term COORDINATED is applied to those parts of the sentence, which are introduced by a coordinating conjunction or punctuation mark.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

Poor Original

... arily.

Through application of 'Rules of Word Selection' in the cour- ... procedure all words of lesser category than the ... are omitted, chief constituents of the grammatical pattern required being thus singled out.

This is not the place to give a detailed discription of all the processes involved in the analysis of verb patterns in Section III. For this reason, the discussion will be restricted to just a few comments on patterns that bring about the most interesting solutions. They are Patterns I:Ia, Ib; and 2:2a.

Among different solutions of Pattern I:Ia noteworthy is that of transforming English construction of Modal Passive,

i.e.	Modal Verb	+	Selected Verb indicated 'Auxiliary I' (BE)	+	Analyzd Verb	.

into Russian Active Compound predicate.

Modal Verb; Impersonal	+	Analyzed Verb, Infinitive,

the transformation being associated with conversion of English subject into Russian Direct object (See Tables Nos.IO and I2).

Pattern 2:2a is provided, among other solutions, with that of transformation of English Complex Object Construction into Russian subordinate ci use.

Resultant characteristics developed for the verbs analyzed in Section III include both morphological and syntactical information. Of syntactical indications only 'Predicate' and 'Attribute' are fixed here, the former being associated with morphological indicadions [illegible] (I[illegible]tive, subjunctive and infinitive are developed here), of tense (Present or Past) and voice (both Active and Passive are here developed).

The indication 'Attribute' is accompanied by morphological indications of 'Participle', tense (Present or Past) and voice (Active or Passive).

ING-forms of the verbs are defined in Section IV, where the same verb patterns are analyzed, though important changes in their value affect the order in which they are searched here.

Pattern I:Ib disappears, whereas Patterns I:Id and 3:3a, 3b, are much wider represented here, the former pattern being complicated by differentiating quite a number of semantic groups of verbs significant for the Analysis. These groups are Nos I to 11, Class I, and Nos 2,7,9,21 and 24, Class II (See below).

There are some differences in modifications of Patterns 4:4a, 4[illegible]. [illegible]o complete the picture, another two patterns should be mentioned, which are here introduced. They are Patterns 3:3c and 5:5a.

The resultant characteristics of the Russian equivalent verb include one of the following sets of indications: I) 'Verbal Noun, Neuter'; 2) 'Participle, Present tense, Active voice; Attribute'; 3) 'Verbal Adverb, Present (or Past) tense; 4) 'Not to be Translated (NT), to be Disregarded' (D). In addition to these, 'Infinitive', 'Subjunctive' or 'Indicative mood', with the corresponding set of indications, is developed in case the analyzed verb takes its characteristics from some of the 'selected (helping) words'.

Verbs not inflected are analyzed in Section V. Verb Patterns I:Ia, Ib, Ic, Id are replaced here by I:Ie, If, Ig; Patterns 2:2b,2c,

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3

Poor Original

are added to Pattern 2:2a, which, in its turn, is greatly enlarged. Patterns 4:4a and 5:5a disappear; instead, Pattern 4:4c increases considerably, and Patterns 5:5b-1, 5b-2, 5b-3 are introduced.

Among different solutions of Patterns 5:5b-I and 5:5b-2 at least those two are worth special mention, which deal with transformation of the English constructions of Complex Subject and of attributive Infinitive into the Russian complex sentence or subordinate clause, accordingly.

The resultant information here includes the indication of Infinitive, Imperative, Subjunctive or Indicative mood, with the indications of tense (Present, Past or Future) and voice (Active or Passive) attached in case of the Indicative mood. The only syntactical indication fixed here is 'Predicate'

Verb analysis in different section of the routine is based on the classification of the verbs, devised to characterize English verbs both within the English system and with regard to the Russian translational traditions.

Within the English language verbs are classified into 'MODAL' and HALF-MODAL (help, dare), AUXILIARY and 7 sub-classes of HALF-AUXILIARIES, CAUSATIVE (cause, enable, make, order, command, etc.), DECLARATIVE (declare, call, label, report, etc.), verbs taking two Objects (give, offer, permit, etc), aso.

To meet the requirements of the Russian translational traditions, verbs are divided into classes and semantic groups. To date, 53 group of verbs have been established. These are summarized into three classes, the first two classes comprising verbs having translational peculiarities in finite (class I) or infinite (class II) forms; class III covers more complicated cases.

Poor Original

The 'Verb Analysis' routine is applied until every verb of the sentence is provided with all the grammatical information required in the synthesis routines, except for the indications of number, person (or gender) which are not defined until the subject of the Russian sentence is established.

Noteworthy is the fact that the information obtained in this routine is not restricted to the analyzed verb, but is extended to cover the information available at this stage of Analysis concerning 'selected' (helping) words (verb, nouns, adjectives, etc.) and punctuation marks. Moreover, quite a number of transformations in sentence structure are introduced here, which include change of word-order, inserting necessary conjunctions and other words or punctuation marks, etc. These transformations are associated with the translation of Complex Subject and Complex Object, Attributive Infinitive and Gerundival Subject, as well as some other verb constructions.

B.2. The 'Verb Analysis' routine is followed by the routine devised to analyze the punctuation marks of the English text with the exception of those terminating the utterance[1/].

The analysis here serves two ends. One is to establish the 'English function' (i.e. a function within the English text) of every punctuation mark, the second aim being their 'Russian func-

---

1/ Note, that our application of the term UTTERANCE is at variance with its usual applications. A piece of text, terminated by full-stops, exclamatory or question marks, we call 'utterance', in order to distinguish it from the SENTENCE, by which only a simple sentence is understood, i.e. a sentence containing not more than one non-coordinated predicate.

tion', by which their Russian correspondents are understood. These functions may or may not coincide. In the latter case, both English and Russian indications are developed. Thus, commas marking out a propositional phrase, obtain English indications CP (Comma, Parenthetical), associated with Russian indications CD_R (Comma, to be Disregarded in Russian), as a result of which this comma will not appear in the Russian text.

There are cases, when English punctuation marks should be neglected in the course of English Analysis, though rendered by the same mark in the Russian text. This is achieved by developing the indication CD (Comma, to be Disregarded in English).

B.3. The 'Syntax Analysis' routines succeed the Analysis of Punctuation Marks, since it is essential that the information which can be obtained in both previous routines should be available here. The analysis is carried out by three cycles.

In Cycle I, Parentheses, comparative AS-phrases and Attributive word-groups, with a participle, verbal adverb or adjective as chief constituent, are marked out by means of appropriate qualification (and insertion, when necessary) of punctuation marks. This qualification includes the development of indications $CP^{b/e}$ (Comma, Parenthetical, beginning/end) and $CA^{b/e}$ (Comma, Attributive, beginning/end).

Attributive groups are not isolated until preliminary check-ups for certain patterns of grammatical context have been carried out. Among these patterns are the following three (See Fig.3):

I. Preceeding word is CA^b

Poor Original

2.	Preceeding word	is	Noun	(on condition that it is associated with Pattern 3)
3.	Following word	is	3a. Preposition; 3b. Conjunction /or Conjunctive Word/ 3c. Punctuation Mark (PM); 3d. Verb indicated 'Participle, Short form'.	(may or may not be associated with Pattern 2).

Fig.3.

Practically, isolation of the above-mentioned word-groups comes to establishing their right boarder ('end'), since the left boarder ('beginning') in these cases can easily be associated with the chief constituent of the construction.

The 'end' of the isolated word-group is searched to the right of the chief constituent until the nearest following

a) CA^{e} ;

or b) Noun with indication (or conditions of) Subject;

or c) VERB with indication Predicate;

or d) Conjunction without indication 'coordinating';

or e) Punctuation Mark without indications CP or 'b' ('beginning'),-

is found. It is essential that the search should be performed in the order indicated above[1/].

In Cycle II sentence boarders are established by checking up the

---

1/ Mind that wherever following or preceding words are searched, 'Rules of Word Selection' are strictly observed in the Analysis routines.

Poor Original

utterance [1/] for the presence of

I) Conjunctions with indication 'Inhomogoneous'(CI);

and/or..... 2) Conjunctive words (nouns or adjectives);

" ..... 3) words with indication 'Initial';

" ..... 4) two (or more) Predicates within a passage terminated by sentence boarders already established;

a) immediately following each other,

b) not immediately following each other;

" ..... 5) two nouns following each other, but not joining in a 'lawful' combination.

A very detailed analysis of every pattern is carried out in the order indicated above. Ordinarily, sentence boarders do not acquire the indications SB (Sentence Beginning) or SE (Sentence End) at this stage of the analysis, but for two cases:

a) when pattern analyzed is

CI 'of condition' + Adjective indicated 'Predicative' + /Absence of Noun/ ;

b) when two conjunctions follow each other, the latter being provided with a correlative conjunction or conjunctive word.

At this stage of the Analysis certain changes in the structure of the Russian text are also provided. In this connection, mention should be made of the insertion of the conjunctive word 'KOTOРЫЙ' (with appropriate indications) in case it is omitted in the English attributive clause (Patterns 4 and 5).

In Cycle III information obtained by this time is used to qualify sentence-boarders as indicating 'Beginning' or 'End' of the sentence inserted within the boarders of another sentence. Among

1/ See above our definition of the term.

Poor Original

other more particular cases, that of sentences where subject and predicate are separated by an attributive or other subordinate clause, is analyzed here. Initial, middle or final position of the nonpredicative piece of the broken sentence is considered decisive for the order in which they are dealt with .

Boarders of the sentences which have not been recognized as 'insertions' within other sentences are qualified as SD (Sentence Division), since neither 'Beginning' nor 'End' indication is considered necessary here.

The information obtained by application of the 'Syntax Analysis' routine is extremely valuable, as long as syntactical units for further Analysis are marked out. Nouns, Numerals and Adjectives are analyzed within these units, the order in which Syntactical units are treated being as indicated below;

I. Sentence (minus all Parenthetical and Comparative or Attributive word-groups);

2. Comparative and Attributive word-groups minus Parenthetical word-groups) within this sentence;

3. Parenthetical word-groups within this sentence;

4. Next Sentence (minus all Parenthetical and Comparative word-groups);

Step 4 is again followed by Steps 2,3,4 aso., till the last sentence is looked through.

.4. The'Noun Analysis' routine is devised so as to cover the analysis of two word-classes, which are Nouns and Numerals. The so-called 'ordinal numerals' being qualified as Adjectives /5/, only cardinal numerals are termed Numerals here. These are not entered into the class of Nouns owing to their morphological peculiarities.

The routine is divided into two parts, the development of CASE indication being the target of Part I, whereas in Part II the indication of NUMBER is developed. The two parts differ in scope as well as in method.

In Part I, where both Nouns and Numerals are treated syntactical methods are used, since qualification of nouns inflected with "'S", " ' " or "IAN" has been achieved at an earlier stage (See Table No 5). Grammatical context of the analyzed Noun or Numeral is checked up for the presence of some 'governing' or 'coordinating' element preceding the analyzed word. Prepositions, verbs, verbal nouns and numerals belong to the 'governing' group, whereas conjunctions and punctuation marks with indication 'Homogeneous' or conjunctions of comparison are considered 'coordinating'. In both cases the indication required is taken from one of the preceding words, either governing or coordinated with the analyzed one. If neither is the case, other patterns are applied. Special attention is given to Conjunctive Nouns.

As to Part II of the routine, Nouns are the only class of words analyzed here, morphological methods providing the most important information for developing the indication of number. If the word is 'inflected'[1/], the number is defined as 'Plural', otherwise syntactical methods are applied.

Mention should be made of the fact, that patterns of grammatical context are solved here so as to reflect the peculiarities of Number and Case forms in the English and Russian languages. Thus,

---

1/ See Notes on Table No 5.

with number indications, differences in classifying nouns into 'countables' and 'uncountables' in English and in Russian are taken into account. Among other idiomatic constructions in Russian that of 'Numeral + Noun' combination, where the Numeral has retained the old 'dual' government, should be pointed out. Certain peculiarities in Russian verb government are also given consideration.

Syntactically, Nouns and Numerals are classified only when used in the function of an Atribute or Subject of the sentence.

<u>B.5. The 'Adjective Analysis' routine</u> comes the last in the series of English Analysis routines which develop grammatical indications to be used in the Russian Synthesis routines. Information acquired of the Adjectives (and Participles) of the text includes the indications of gender, number, case, degree of comparison and short/full form. In addition to these, the indications of 'Substantivized' or 'Adverbial Adjective' are developed.

Preliminary ckeck-ups for the absence of indications required are followed by testing the morphological structure and syntactical environment of the analyzed word. The main interest of the testing procedure lies in finding out whether the analyzed word is placed in the attributive position towards some noun of the sentence1/. If the search is positivo, it becomes very important to pick up the right noun, which in some cases is not a very easy task.

Another important search is aimed at establishing a predicative position of the analyzed word. Finally, if the search is negative,

---

1/ Sometimes it can be a noun of another sentence, as in the case of conjunctive adjectives.

the word is qualified as '*Substantivized*'.

<u>B.6. The 'Changes of Word-Order' routine</u> is meant to give a 'final touch' to the translated text before the *Synthesis* routines are applied.

English patterns of word-order which do not correspond to Russian patterns are recomposed. It is remarkable, however, that these recompositions are mostly of local character.

The most important changes of word-order, perfomed in accordance with this routine[1/], are due to the difference in the position of attributes expressed by nouns or noun combinations(See Tables No 11 and 12) as well as in the expression of negation in the English and Russian languages.

Other changes are of no particular importance.

<u>4. Conclusions.</u>

The heart of the whole method suggested above lies in the most carefu[illegible] of every language included in the MT system, a very detailed subsequent comparison of these discriptions being the basis of MT research.

The comparison of the English and Russian languages in the course of MT studies has proved to be more fruitful than could have been supposed, insofar as the structure of these languages has been found strikingly alike, up to a great many details. For this reason, an attempt was made to work out an Anglo-Russian MT scheme where maximum similiarities found in the structures of the two languages

1/ Certain more specialized changes of word-order are performed at an earlier stage of Analysis (See above: Sections B.I. and B.2).

Poor Original

ould be made use of.

Owing to this, structural transformations of the translated text have been restricted in the present scheme of MT to such minimum as omittance and insertion of just a few 'helping' words or punctuation marks and a few (local) changes of word-order. Nevertheless, the translations thus obtained are quite adequate for understanding and do not require post-editing, as can be seen in the samples cited below (See Tables Nos 1,2,3 and 4).

Poor Original

References.

I. D. Panov "Concerning the Problem of Machine Translation of Languages", Moscow 1956.

2. I.S. Mukhin. "An Experiment of the Machine Translation of Languages Carried Out on the BESM".

3. I.K. Belskaja. "Machine Translation of Languages", RESEARCH, vol.10 (October 1957) pp.383-389.

4. ТЕЗИСЫ КОНФЕРЕНЦИИ ПО МАШИННОМУ ПЕРЕВОДУ. МОСКВА, 1958.

5. СБОРНИК СТАТЕЙ ПО МАШИННОМУ ПЕРЕВОДУ. ИТМ и ВТ АНСССР (В ПЕЧАТИ).

[illegible] Institute of Computing Machinery

Candidate of Technical Sciences P.T. Slichuk;
P.T. Dobrosmyslov.

STAT

ALGORITHMS AND MACHINE LOGIC FOR SOLVING PROBLEMS CONNECTED WITH THE ECONOMICAL ANALYSIS OF THE ACTIVITY OF MANUFACTURING PLANTS

3.

1. The analysis of the activity and management of an industrial enterprise is based on the study of information obtained both in the course of manufacturing processes and from outside.

As a rule, the information "word" comprises two elements: the classification group and the base. The first defines the information economic attributes, while the second is a concrete or an abstract number.

2. Accounting and planning algorithms are obtained by means of complex operations in which are involved both the classification groups and bases. These operations are limited to a comparatively small set of standard operators.

3. The standardisation of operators allows to optimize the design and characteristics of the machines for the automation of economic estimates and calculations taking into consideration the scope of processed information and the predetermined terms for obtaining the required results, as well as the reliability, the cost and the service conditions of the machine.

4. The realization of the afore mentioned points is illustrated by the example of a machine intended, for economic analysis, which is at the present time under development.

5. The economic efficiency of a machine is determined not only by the reduction of labour expended in processing information but also by the effect obtained in reducing the lag in time in the data processing circuit.

6. Further development of means of automation for management control at manufacturing plants necessiates the use of mathematic schemes, these being adequate to actual management systems, and of effective algorithms ensuring realization of an optimum management process. Moreover, it is necessary to proceed with futher investigations on the economic efficiency of machines intended for the automatic management control of industrial plants.

Declassified in Part - Sanitized Copy Approved for Release @ 50-Yr 2014/02/27 : CIA-RDP81-01043R003400230002-3